Neural Implementation of (Approximate) Bayesian Inference

Michael Rescorla

§1. Bayesian modeling of perception

In recent decades, Bayesian modeling has achieved extraordinary success within perceptual psychology (Knill and Richards, 1996; Rescorla, 2015; Rescorla, 2020a; Rescorla, 2021). Bayesian models posit that the perceptual system assigns *subjective probabilities* (or *credences*) to hypotheses regarding distal conditions (e.g. hypotheses regarding possible shapes, sizes, colors, or speeds of perceived objects). The perceptual system deploys its subjective probabilities to estimate distal conditions based upon proximal sensory input (e.g. retinal stimulations). It does so through computations that are fast, automatic, subpersonal, and inaccessible to conscious introspection.

More formally, the perceptual system maintains a *prior probability* $p(h)$, where each h is a different hypothesis about distal conditions. The perceptual system also maintains a *prior likelihood* $p(e \mid h)$ that assigns a probability to sensory input *e conditional on h* (e.g. the probability of receiving retinal input e given that a perceived object has a certain size and is located a certain distance away). Upon receiving input e, the perceptual system computes the *posterior probability* $p(h \mid e)$. Bayes's Theorem expresses the posterior in terms of the prior probability and the prior likelihood:

$$p(h \mid e) = k\, p(h) p(e \mid h),$$

where k is a normalizing constant to ensure that probabilities sum to 1. The posterior assigns a probability to *h conditional on sensory input e*. Based on the posterior, the perceptual system

selects a privileged estimate h^* that goes into the final percept. In many Bayesian models, though not all, the privileged estimate h^* is the *maximum a posteriori* (MAP) hypothesis:

$$h^* = \text{argmax}_h\, p(h \mid e).$$

The privileged estimate h^* is usually accessible to conscious introspection. In contrast, the priors and the posterior are not typically consciously accessible. Neither are the computations that convert the priors into the posterior or that select h^*.

Bayesian models supply satisfying explanations for numerous perceptual phenomena. A good example is the motion estimation model given by Weiss, Simoncelli, and Adelson (2002). The model posits a "slow motion" prior, i.e. a prior that favors slow speeds. Citing Bayesian inference based on this prior, the model explains a host of motion illusions that had previously resisted unified explanation. Thanks to such explanatory achievements, the Bayesian framework now enjoys orthodox status within perceptual psychology.

A natural question raised by Bayesian perceptual psychology is how the brain implements Bayesian inference. How do neural states physically realize the priors and the posterior? Which neural operations effectuate the transition from priors to posterior? These questions have been intensively studied in computational neuroscience, and there are now several proposed neural implementation mechanisms. One proposal, well known to philosophers through the work of Clark (2015) and Hohwy (2014), highlights a computational strategy known as *predictive coding*. Other proposals, less known to philosophers, do *not* feature predictive coding.

This paper canvasses several proposed implementation mechanisms, including both predictive coding and alternatives. I will not try to provide anything like an adequate survey. Nor will I defend one approach over another. Instead, I aim to promote an enhanced appreciation

within the philosophical community for the diverse neural implementation mechanisms currently under active investigation. Reflection on diverse candidate neural implementation mechanisms offers several benefits. First, and most obviously, we gain a more comprehensive vista on current computational neuroscience. Second, we elucidate what it means to attribute subjective probabilities to the perceptual system. Third, we clarify the sense in which the perceptual system may be said to execute Bayesian inferences.

§2 presents background material on Bayesian inference in physical systems. §§3-4 reviews various proposals for neural implementation of Bayesian inference. §5 compares the proposed implementation schemes with neural networks that *simulate* Bayesian inference. §6 explores the methodological implications of my discussion.

§2. Credal states and transitions

A Bayesian model posits *credal states*: assignments of credences to hypotheses. It also posits *credal transitions*: transitions among credal states. The simplest models posit a single credal transition from the prior probability and the prior likelihood to the posterior. More complex models posit iterated credal transitions in response to sequential new sensory input. For example, the object-tracking model in (Kwon, Tadin, and Knill, 2015) posits iterated credal updates regarding the position and velocity of a moving stimulus.

Elsewhere, I have defended a *realist* view of Bayesian perceptual psychology (Rescorla, 2020b). Realism holds that, when a Bayesian perceptual model is empirically successful, we have reason to believe that the model is approximately true. More specifically: when a Bayesian perceptual model is empirically successful, we have reason to believe that there are credal states and transitions resembling those posited by the model. For example, the empirical success of the

motion estimation model provides reason to hold that human motion estimation deploys a "slow motion" prior similar to that posited by the model. The model's theoretical apparatus corresponds at least roughly to psychological reality.

Block (2018), Colombo and Seriès (2012), Orlandi (2014), and others espouse an opposing *instrumentalist* perspective. According to instrumentalists, empirical success of a Bayesian perceptual model provides no reason to believe that the perceptual system executes anything resembling the computations posited by the model. We may only conclude that the perceptual system operates *as if* it executes those computations. More specifically, we have no reason to posit that perception involves credal states or transitions. A Bayesian model is just a useful predictive device that helps us summarize input-output mappings. For example, the motion estimation model specifies a mapping from retinal inputs to motion estimates. According to instrumentalists, the model tells us nothing about the mental processes that mediate between inputs and outputs, save that the processes generate the specified input-output mapping.

To clarify the debate between realism and instrumentalism, I elucidate *credal states* in §2.1 and *credal transitions* in §2.2. In §3, I build upon those elucidations to address how the brain might implement credal states and transitions.

§2.1 Implicit encoding of credences

How might a physical system encode an assignment of credences to hypotheses?

The most straightforward encoding scheme is *explicit enumeration*: the system explicitly lists the credence assigned to each hypothesis. Unfortunately, enumeration is not feasible when the hypothesis space is infinite, as it is in most serious scientific applications.

An alternative scheme is *parametric encoding*: the system encodes a probability distribution through a few parameters. Many examples of parametric encoding involve a *probability density function* (pdf): a nonnegative function $p(x)$ over $\mathbb{R}$ whose integral is 1. We derive a probability distribution from a pdf through integration: the probability assigned to interval $[a, b]$ is the integral of the pdf over the interval $[a, b]$. See Figure 1. Often, although not always, one can encode a pdf through a few parameters. A familiar example is the family of Gaussian distributions. Each Gaussian has a pdf of the form:

(1) $$p(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{-(x-\mu)^2}{2\sigma^2}} .$$

See Figure 2. Here μ is the mean of the distribution and σ^2 is the variance. We can encode a Gaussian through the parameters (μ, σ^2).

INSERT FIGURES 1 AND 2 ABOUT HERE

When a probability distribution is not finitely parametrizable, another encoding scheme is needed. One widely used encoding scheme involves *sampling*. Consider a system that draws samples from the hypothesis space. We can delineate an objective chance function $c(h)$ that governs the system's sampling behavior. In the simplest case, $c(h)$ is the objective chance that the system draws sample h. In more complicated cases, $c(h)$ may instead be a density that determines objective chances through integration. Either way, $c(h)$ specifies the system's sampling propensities over the hypothesis space. As several researchers have proposed (Fiser, et al., 2010; Icard, 2016; Sanborn and Chater, 2016), sampling propensities can serve as subjective probabilities. We may delineate a subjective probability assignment via the equation

$p(h) = c(h)$,

where the right-hand side specifies an *objective* probability (or probability density) and the left-hand side specifies the encoded *subjective* probability (or probability density). Sampling encoding is widely used in statistics (Gelman et al. 2014) and machine learning (Murphy, 2012).

Crucially, parametric and sampling encoding are implicit rather than explicit. When a system encodes a Gaussian through the parameters (μ, σ^2), the system does not explicitly enumerate any credences. Instead, credences are implicit in the specification of μ and σ^2 (on the understanding that the encoded distribution is a Gaussian). Similarly for sampling encoding: credences are implicit in the system's sampling propensities.

Implicit encoding of probabilities is crucial for understanding the debate between realism and instrumentalism. The credal states posited by Bayesian perceptual psychology are typically defined over an infinite (indeed, uncountably infinite) hypothesis space. For example, the set of possible speeds is uncountable, so the motion estimation model is defined over an uncountable hypothesis space. When the hypothesis space is infinite, explicit enumeration of credences is not an option. Any plausible realist position must acknowledge that the perceptual system typically encodes credences implicitly rather than explicitly. Credences may be encoded through a parametric scheme, a sampling scheme, or some other scheme.

Given that credences can be encoded in such diverse ways, we naturally ask *why* a physical state counts as encoding credences. What do all possible encoding schemes have in common such that they count as encodings of credences? A truly satisfying answer would give non-circular necessary and sufficient conditions for a physical system to assign a credence to a hypothesis. Beginning with Ramsey (1931), there have been several attempts to supply the desired necessary and sufficient conditions. Unfortunately, these attempts are now widely

regarded as problematic (Erikkson and Hájek, 2007). As a result, we cannot say what it is for a physical state to realize a credal state. Nevertheless, we can assert with great confidence that credences are physically encoded in diverse ways. After all, parametric and sampling encodings are used on a daily basis in practical applications of the Bayesian framework.

§2.2 Computational intractability of Bayesian inference

Sometimes, it is easy to compute the posterior from the prior probability and the prior likelihood. To illustrate, suppose that the prior is a Gaussian of the form (1) and that the prior likelihood has the Gaussian form:

(2) $$p(y \mid x) = \frac{1}{\sqrt{2\pi}\tau} e^{\frac{-(y-x)^2}{2\tau^2}}.$$

An idealized Bayesian agent who starts with these priors will respond to sensory input y by forming new credences given by the posterior $p(x \mid y)$. One can show that the posterior $p(x \mid y)$ is a Gaussian with mean η and variance ρ^2 given by

(3) $$\eta = \frac{\frac{\mu}{\sigma^2} + \frac{y}{\tau^2}}{\frac{1}{\sigma^2} + \frac{1}{\tau^2}}$$

$$\frac{1}{\rho^2} = \frac{1}{\sigma^2} + \frac{1}{\tau^2}.$$

See (Gelman et al., 2014, pp. 39-41) for details. The posterior mean η is a weighted average of the prior probability mean μ and the fixed value y, with weights inversely proportional to the respective variances. Intuitively, then, η is a compromise between the prior probability and the sensory input y.

To obtain a helpful visualization of (3), we can hold y fixed and note that (2) then yields a one-place function of x:

$$L(x) = p(y \mid x).$$

$L(x)$ is called a *likelihood function*. The posterior

$$p(x \mid y) = k\, p(x)p(y \mid x) = k\, p(x)L(x)$$

is found by multiplying the prior $p(x)$ with the likelihood $L(x)$ and normalizing. See Figure 3.

INSERT FIGURE 3 ABOUT HERE

Computing the posterior is not usually as easy as in (3). A neat self-contained description of the posterior may not exist. Even when a self-contained description exists, finding it may require computational resources beyond those available to a realistic agent (Kwisthout, Wareham, and van Rooij, 2011). Specifically, calculating the normalizing constant k in Bayes's theorem may be a computationally intractable task.[1] In general, then, a physical system with limited time and memory may not be able to compute the posterior from the priors.

The standard solution within Bayesian decision theory is to settle for *approximate* Bayesian inference. Even when Bayesian inference is computationally intractable, there may be a tractable algorithm that comes close. There are two main approximation strategies:

- *Variational algorithms* approximate the posterior using a probability distribution drawn from a nicely behaved family (e.g. Gaussian distributions). The basic idea is to pick the distribution from this family that is "closest" to the actual posterior.

[1] Roughly speaking, a computation is *tractable* when it can be executed by a physical system with limited time and memory at its disposal. A computation is *intractable* when it is not tractable. For discussion of computational tractability in relation to cognitive science, see (van Rooij et al., 2019).

- *Sampling algorithms* approximate the posterior by drawing samples from the hypothesis space. In response to input *e*, the system alters its sampling propensities regarding each hypothesis *h*.

In both cases, the physical system instantiates credal states and transitions. It begins with a prior probability $p(h)$ and a prior likelihood $p(e \mid h)$. In response to input *e*, it transitions to a new credal state $p_{new}(h)$ that approximates the posterior $p(h \mid e)$. The relevant credal assignments may be implicit rather than explicit. For example, the posterior may be encoded by sampling propensities. See (Murphy, 2012) for detailed discussion of variational and sampling approximation algorithms.[2]

Bayesian perceptual models commonly posit priors that support tractable Bayesian inference. However, the human perceptual system need not instantiate such mathematically convenient priors. For example, numerous perceptual models posit Gaussian priors, but we know that the human perceptual system sometimes uses priors that with heavier tails than Gaussians (Stocker and Simoncelli, 2006). As Bayesian perceptual psychology develops, it will doubtless assign greater prominence to approximate rather than exact Bayesian inference. An example is the model of binocular rivalry given by Gershman, Vul, and Tenenbaum (2012), which posits a sampling approximation to an intractable Bayesian inference. The model explains a range of perceptual phenomena that arise during binocular rivalry, such as the distribution of switching times among percepts.

---

[2] For some sampling algorithms, it is natural to view credences as encoded by the distribution of samples rather than by sampling propensities. Suppose that the algorithm draws *n* samples $v_1, v_2, \ldots, v_n$. For $i \leq n$, let δ_{v_i} be the Dirac measure centered at v_i (i.e. the measure that allocates all probability mass at v_i). We may regard the samples as encoding a probability distribution *q* via the equation $q = \frac{1}{n}\sum_{i=1}^{n}\delta_{v_i}$. This encoding scheme figures in the *particle filter*, which approximates iterated Bayesian inference given sequential inputs (Crisan and Doucet, 2002; Murphy 2012, pp. 825-837). At each time, the particle filter responds to new input by drawing a new set of *n* samples; the new samples encode a new probability distribution *q*. As the number of samples goes to infinity, the distribution *q* encoded at each time converges to the true posterior at that time.

My realist viewpoint extends straightforwardly to perceptual models that postulate approximate Bayesian inference. Realism holds that, when an approximately Bayesian model is empirically successful, we have reason to hold that the model is approximately true. We have reason to hold that the perceptual system instantiates credal states and transitions resembling those postulated by the model.

Block (2018) suggests that the intractability of Bayesian inference poses a problem for realism about Bayesian perceptual psychology. Once realists concede that the perceptual system executes approximate Bayesian inference rather than exact Bayesian inference, how does their position ultimately differ from instrumentalism? As Block (2018, p. 8) puts, "[W]hat is the difference between approximate implementation of Bayesian inference and behaving roughly as if Bayesian inference is being implemented…? Until this question is answered, the jury is out on the dispute between realist and anti-realist views."

My reply: there is a huge difference between physical systems that approximately execute Bayesian inference and physical systems that merely behave *as if* they approximately execute Bayesian inference. A system that approximately executes Bayesian inference instantiates credal states and transitions:

- The system begins with a prior probability and a prior likelihood.
- In response to sensory input *e*, the system transitions to a new credal state that approximates the posterior.
- The system can then deploy the approximate posterior in further computation, such as selection of a privileged estimate *h**.

See Figure 4. In contrast, a system that merely behaves *as if* it approximately executes Bayesian inference need not instantiate any credal states. For example, a system might simulate Bayesian

estimation through a (very large!) look-up table. In response to input *e*, the look-up table system selects an output *h** close to the output that a Bayesian estimator or approximately Bayesian estimator would select. The look-up table system does not instantiate any credal states, let alone credal transitions. A physical system that approximately executes Bayesian inference has a different internal causal structure than a system that merely simulates approximate Bayesian inference (Rescorla, 2020b).

INSERT FIGURE 4 ABOUT HERE

The difference in causal structure has important methodological implications. Realists posit credal states embedded in the causal structure depicted by Figure 4. Since mental states and processes are physically realized in the brain, it becomes a pressing task to investigate how credal states and transitions are neurally realized. We must illuminate how neural activity implements the causal structure depicted by Figure 4. From a realist perspective, the search for neural implementation mechanisms of (approximate) Bayesian inference looks like a vital research endeavor (Ma, 2019). From an instrumentalist perspective, we have no reason to suspect that the brain implements a causal structure remotely like Figure 4, so we have no reason to take Figure 4 as a guide to neural mechanisms. If there are no credal states and no credal transitions, then it is a waste of time to investigate how credal states and transitions are neurally realized.

Evidently, the dispute between realism and instrumentalism is not just an abstract "philosophical" debate about how to interpret a fixed scientific theory. The dispute has major implications for which research avenues in neuroscience look promising and which do not. My goal for the rest of the paper is to gain insight into these methodological implications and,

thereby, into the dispute between realism and instrumentalism. In §3, I examine some neural network models that implement approximate Bayesian inference. The models vary in biological plausibility, but some of them are under active consideration by computational neuroscientists. In §4, I examine neural network models that merely *simulate* approximate Bayesian inference. By comparing the models from §3 with the models from §4, I hope to clarify the diverging theoretical and methodological commitments of realism and instrumentalism.

§3. Some proposed neural implementation schemes

There are several elements we should expect from any complete theory of how the brain implements approximate Bayesian inference.

To begin, our theory will identify a neural variable U that encodes the prior probability $p(h)$. Each value u of U corresponds to a possible neural state (e.g. a profile of firing rates across a neural population). U's value determines the prior, assuming appropriate background conditions. So U satisfies counterfactuals of the form:

(4) If neural variable U were to have value u in background conditions B, then the perceptual system would assign credence $p(h)$ to h.

In principle, different values of u may encode the same prior. In practice, different values of u usually encode different priors.

The qualifier regarding background conditions B is crucial because neural state *taken on its own* does not usually determine credal state. A credal state assigns subjective probabilities to hypotheses. In the case of perception, the hypotheses concern distal properties, such as shape, size, color, location, speed, and so on. The brain represents distal properties only due to causal relations that it bears to the distal environment (Burge, 2007; Burge 2010), perhaps lying within

the organism's developmental history or its evolutionary past. The requisite causal relations do not supervene upon internal neurophysiology. In principle, an organism with identical neurophysiological properties could be embedded differently in the physical world, bearing such different causal relations to the distal environment that it represents different distal properties (Burge, 2007; Burge, 2010; Egan, 2010). Different represented properties entail a different credal state --- an assignment of credences to different hypotheses regarding different distal properties. Since credal state does not supervene upon internal neurophysiology, it would be futile to seek a neural variable that determines credal state *on its own*. The best we can do is find a neural variable that determines credal state *assuming certain background conditions*, including certain causal relations to the distal environment.

Ultimately, we would like to illuminate the assumed background conditions. What background conditions must obtain for neural state u to guarantee that a given credence is assigned to h? Answering that question would require progress towards necessary and sufficient conditions for physical realization of credal states. It would also require progress on *the problem of intentionality*, i.e. the problem of what it is for mental states to have representational properties (Loewer, 1997). Fortunately, scientific theorizing about neural implementation mechanisms need not await progress on these deep questions. Lots of research in computational neuroscience addresses neural implementation while tacitly assuming whatever background conditions are needed to ensure suitable counterfactuals (4).

Formally speaking, we may summarize the connection between priors and neural states through an equation of the form:

(5) $p(h) = \Phi(u)$,

where $p(h)$ is either a probability distribution or a pdf; u is a possible value of neural variable U; and Φ is a function that carries each u to $p(h)$.[3] Φ is sometimes called a *decoder*: it shows how to "decode" the probabilistic import of a neural state.

A complete implementation theory will also address how prior likelihoods are encoded. It will identify a neural variable V that satisfies counterfactuals of the form:

(6) If neural variable V were to have value v in background conditions B, then the perceptual system would assign credence $p(e \mid h)$ to e conditional on h.

In parallel to (5), we may formalize the encoding through an equation of the form:

(7) $p(e \mid h) = \Psi(v)$.

In practice, different values of V usually encode different prior likelihoods.

A complete implementation theory will additionally specify how the brain responds to input e. Here we must distinguish between *deterministic* versus *stochastic* transitions. In the deterministic case, we want a function f of the form

$$w = f(u, v, e),$$

where w is a possible value of some neural variable W. In the stochastic case, we want the chance distribution governing the transition from u, v, and e to w. Whether deterministic or stochastic, our model will describe operations that transform u, v, and e into w. The operations must be biologically plausible, i.e. real neural populations must be able to execute them.

Since our goal is to model credal transitions, each value w of W will encode the new credal state induced by sensory input e. So W will satisfy counterfactuals of the form:

[3] I intend the left-hand side of (5) to denote a *function* that assigns a probability (or a probability density) to each possible h. To be more careful, I should use lambda notation and write (5) as $\lambda h.p(h) = \Phi(u)$. Similarly for equations (7) and (9) below. However, lambda notation seems to me needlessly fussy for present purposes. Throughout the text, I sloppily use the expression "$p(h)$" sometimes to denote a function and sometimes to denote a specific real number assigned to a specific h. Context should make clear which denotation I have in mind.

(8) If neural variable W were to have value w in background conditions B, then the perceptual system would assign credence $p_{new}(h)$ to h.

In parallel with (5) and (7), we can formalize the encoding through an equation of the form:

(9) $p_{new}(h) = \Gamma(w)$.

The decoder Γ specifies the new credal state corresponding to each w. When the system executes exact Bayesian inference, we have

$$p_{new}(h) = p(h \mid e),$$

i.e. the new credal state is the true posterior. In general, the system may only approximate the posterior, in which case we have

$$p_{new}(h) \approx p(h \mid e).$$

In practice, different choices of e usually induce different values of W, and different values of W usually encode different credal states $p_{new}(h)$.

Figure 5 visualizes how the various components of a complete implementation theory fit together. A major goal of contemporary computational neuroscience is to identify decoders and neural operations satisfying something like Figure 5. Researchers pursue that goal by constructing *neural networks*: simplified models of how idealized neural populations evolve. The neural networks vary greatly in their biological realism, but at least some of them are fairly realistic. A suitable neural network, coupled with suitable decoders Φ, Ψ, and Γ, models how the brain might implement approximate Bayesian inference.[4]

[4] Some readers may worry that Figure 5 suggests a problematic "causal overdetermination," whereby credal assignment $p(h)$ and neural state u overdetermine privileged estimate h^*. I want to resist any such interpretation of Figure 5. There is really just one channel of causal influence described twice over: at the psychological level (by citing the credal assignment) and at the neural level (by citing neural state u). The neural level realizes the psychological level, so there is no causal overdetermination. The literature offers several avenues for developing this intuitive diagnosis in more rigorous terms. See (Bennett, 2007; Rescorla, 2014; Woodward, 2008; Woodward, 2015) for discussion of the complex interrelations between mental causation and neural realization.

INSERT FIGURE 5 ABOUT HERE

I will now examine some specific neuroscientific models along these lines. My aims are conceptual rather than empirical: I want to highlight the diverse ways in which credal states and transitions might in principle be neurally realized. For that reason, I will not address evidence for or against the neuroscientific models I discuss.

§3.1 Probabilistic population codes

Certain neurons preferentially respond to specific values of a perceived stimulus (Dayan and Abbott, 2005, pp. 14-16), such as the orientation of a bar. We may associate each such neuron with a *tuning curve* $f_i(x)$, which summarizes the average response of neuron i to stimulus value x. A common posit that fits many neurons fairly well is that $f_i(x)$ is an unnormalized Gaussian. See Figure 6, which depicts Gaussian tuning curves for a population of neurons tuned to a one-dimensional distal variable. Each tuning curve peaks at a preferred value of the variable.

INSERT FIGURE 6 ABOUT HERE

The core idea behind *probabilistic population codes* (PPCs) is that, when a neural population is tuned to distal variable X, the firing profile over the population can encode a probability distribution over X (Knill and Pouget, 2004; Pouget, Dayan, and Zemel, 2003). Figure 7 illustrates. The horizontal axis groups neurons according to preferred values of X. The vertical axis gives each neuron's firing rate on some occasion in response to some fixed stimulus. Let the neural population contain n neurons. r_i is the firing rate for the neuron with preferred

value x_i. $\boldsymbol{r} = \langle r_1, r_2, \ldots, r_n \rangle$ is the profile of firing rates over the population. A particularly straightforward decoder is

(10) $$p(x_i) = \frac{r_i}{\sum_{j=1}^{n} r_j},$$

so that probabilities are concentrated at the preferred values x_i in proportion to the corresponding firing rates and are 0 elsewhere. (The denominator is a normalization constant.) It is often desirable to smooth out the credal assignments so as to avoid concentration of credal mass at preferred values x_i. This can be done through a more sophisticated decoder of the form

(11) $$p(x) = \frac{\sum_{j=1}^{n} r_j \phi_j(x)}{\sum_{j=1}^{n} r_j},$$

where ϕ_j is a pdf associated with neuron j (Zemel, Dayan, and Pouget, 1998). Intuitively, neuron j votes for its preferred pdf ϕ_j with weight proportional to its firing rate. Collectively, the firing rates encode the pdf $p(x)$ defined by (11). Decoders (10) and (11) can be used for likelihood functions or approximate posteriors, with $L(x)$ or $p_{new}(x)$ replacing $p(x)$.

INSERT FIGURE 7 ABOUT HERE

To see (10) in action, consider a neural network with three neural populations N_1, N_2, and N_3. Neural population N_1 responds to sensory input y with firing rate profile $\boldsymbol{r}$. These responses encode likelihood function $L(x)$ via the decoder:

$$L(x_i) = \frac{r_i}{\sum_{j=1}^{n} r_j}.$$

Neural population N_2 has firing rate profile $\boldsymbol{s}$, to which we apply the decoder:

$$p(x_i) = \frac{s_i}{\sum_{j=1}^{n} s_j}.$$

The neural network multiplies firing rates in N_1 and N_2 to determine firing rates in N_3. The firing rate profile $\boldsymbol{t}$ over N_3 is

$$\boldsymbol{t} = < r_1 s_1, r_2 s_2, \ldots, r_n s_n >.$$

If we use the decoder,

$$p_{new}(x_i) = \frac{t_i}{\sum_{j=1}^{n} t_j},$$

then we have

$$p_{new}(x_i) = \frac{r_i s_i}{\sum_{j=1}^{n} r_j s_j}.$$

So the network implements (normalized) multiplication of a prior with a likelihood function. See Figure 8, and see (Gershman and Beck, 2017; Knill and Pouget, 2004) for discussion. This analysis can be extended to decoder (11), though the needed neural operations are more complicated than multiplication of firing rates (Barber et al., 2003; Pouget et al., 2003; Zemel and Dayan, 1997).

INSERT FIGURE 8 ABOUT HERE

One element missing from Figure 8 is encoding of the prior likelihood. In Figure 8, N_1 encodes a likelihood function $L(x)$. Encoding a likelihood function $L(x)$ is not the same as encoding a prior likelihood $p(y \mid x)$: $L(x)$ is a function of x, while $p(y \mid x)$ is a function of x and y.

Nothing I have said addresses realization of the two-place function $p(y \mid x)$. Consequently, I am not inclined to say that Figure 8 depicts genuine Bayesian *inference* (a transition from the prior probability, the prior likelihood, and sensory input to the posterior), although it certainly depicts Bayesian *computation* (computing the normalized product of a prior and a likelihood function).

In a notable contribution, Ma et al. (2006) exploit the stochastic nature of neural responses to analyze how prior likelihoods are encoded. Neural response to a stimulus is governed by an objective chance distribution. More formally, there is a conditional distribution

$$c(\boldsymbol{r} \mid x),$$

where $\boldsymbol{r}$ is firing activity over a neural population and x is the stimulus. Although $c(\boldsymbol{r} \mid x)$ is an *objective* chance distribution, we may regard it as encoding a *subjective* probability distribution. The decoder for the prior likelihood then has the form:

(12) $p(\boldsymbol{r} \mid x) = c(\boldsymbol{r} \mid x)$.

On this approach, stochastic firing propensities of the neural population encode conditional credences (Echeveste and Lengyel, 2018).

A widely used posit, which fits the neurophysiological data fairly well, is that neuron i samples from a Poisson distribution with mean determined by tuning curve f_i and stimulus x:

$$c(r_i \mid x) = \frac{f_i(x)^{r_i} e^{-f_i(x)}}{r_i!},$$

where r_i is the spike count of neuron i during a fixed time interval. Assuming the Poisson distributions independent of one another, we may write

(13) $$c(\boldsymbol{r} \mid x) = \prod_i \frac{f_i(x)^{r_i} e^{-f_i(x)}}{r_i!},$$

where $\boldsymbol{r}$ is the profile of spike counts over the population. Other choices for $c(\boldsymbol{r} \mid x)$ are possible. Assuming (13), a neural population governed by decoder (12) encodes the prior likelihood:

(14) $$p(\boldsymbol{r} \mid x) = \prod_i \frac{f_i(x)^{r_i} e^{-f_i(x)}}{r_i!}.$$

Different choices for $c(\boldsymbol{r} \mid x)$ will yield different encoded prior likelihoods.

(14) has an important virtue: it can support genuine Bayesian inference (Ma et al., 2006). Suppose that the prior likelihood is encoded by neural population N_1 via the decoder (14), and assume that all tuning curves are Gaussians with variance σ_{tc}^2. Holding $\boldsymbol{r}$ fixed, one can show under mild assumptions that the likelihood $p(\boldsymbol{r} \mid x)$ is a (possibly unnormalized) Gaussian with mean y and variance τ^2 given by

(15) $$y = \frac{\sum_{i=1}^{n} r_i x_i}{\sum_{i=1}^{n} r_i}$$

$$\frac{1}{\tau^2} = \frac{\sum_{i=1}^{n} r_i}{\sigma_{tc}^2}.$$

y is a weighted average of the preferred values x_i, with weights given by the spike counts r_i. τ^2 is inversely proportional to aggregate activity in N_1: more spike counts entail lower variance. We may encode the prior probability in the spike count profile for a separate neural population N_2 and the posterior in the spike count profile for a third neural population N_3. Let $\boldsymbol{s}$ be the spike count profile for N_2. Consider a decoder that maps $\boldsymbol{s}$ to a Gaussian prior $p(x)$ with mean μ and variance σ^2 given by

(16) $$\mu = \frac{\sum_{i=1}^{n} s_i x_i}{\sum_{i=1}^{n} s_i}$$

$$\frac{1}{\sigma^2} = \frac{\sum_{i=1}^{n} s_i}{\sigma_{tc}^2}.$$

Applying (3) to (15) and (16), one can easily show that the posterior $p(x \mid \boldsymbol{r})$ is a Gaussian with mean η and variance ρ^2 given by

$$\eta = \frac{\sum_{i=1}^{n} (r_i + s_i) x_i}{\sum_{i=1}^{n} (r_i + s_i)}$$

$$\frac{1}{\rho^2} = \frac{\sum_{i=1}^{n} (r_i + s_i)}{\sigma_{tc}^2}.$$

Thus, the neural network can compute the posterior by adding together spike counts in N_1 and N_2 to determine spike counts in N_3. The spike count profile $\boldsymbol{t}$ over N_3 is

(17) $\boldsymbol{t} = < r_1+s_1, r_2+s_2, \ldots, r_n+s_n >,$

and the encoded Gaussian has parameters given by

(18) $$\eta = \frac{\sum_{i=1}^{n} t_i x_i}{\sum_{i=1}^{n} t_i}$$

$$\frac{1}{\rho^2} = \frac{\sum_{i=1}^{n} t_i}{\sigma_{tc}^2}.$$

See Figure 9. This analysis can be extended beyond Gaussians to a more general family of parametrized distributions (Beck et al., 2007; Ma et al., 2006; Sokoloski, 2017). In the general case, the posterior's parameters are given by linear combination of spike counts rather than by mere addition.

INSERT FIGURE 9 ABOUT HERE

As Ma et al. (2006, p. 1435) note, one potential disadvantage of their model is that the encoded prior $p(x)$ varies across trials due to the stochastic nature of neural firing. The decoder (16) is not well-suited to situations where a stable prior persists across many trials.

An alternative decoder proposed by Ganguli and Simoncelli (2014) avoids this problem and also dispenses with a separate neural population for the prior. We still consider a population N whose neural responses conform to (13). The prior likelihood is still encoded via (14). Rather than posit a separate population that encodes the prior, Ganguli and Simoncelli posit that relatively stable properties of N itself encode the prior.

The core intuition underlying their model is that more neural resources should be associated with more probable stimulus values. An optimally efficient allocation of neural resources will not feature tuning curves spread homogenously across all possible stimulus values (as they are in Figure 6). Rather, tuning curves will be arranged so that preferred values x_i are clustered more densely around probable values of the stimulus. This promotes accurate encoding of more probable stimulus values while downgrading accurate encoding of less probable stimulus values. Ganguli and Simoncelli formalize these intuitions with a *tuning curve density function* $d(x)$, which governs the allocation of tuning curves across the neural population: higher density around x entails more neurons whose preferred stimulus value is near x. Under mild assumptions, each tuning curve f_i can be written as:

(19) $$f_i(x) = k\, f\,(D(x) - i)\,,$$

where k is a constant that modulates maximum average firing rate; f is a fixed function, such as an unnormalized Gaussian, that peaks at 0; and $D(x)$ comes from integrating $d(x)$. f serves as a

tuning curve template. $d(x)$ warps the template as described by (19), yielding a tuning curve f_i with preferred value $D^{-1}(i)$. Ganguli and Simoncelli show that, according to a natural criterion of optimality, the optimal density function satisfies the equation

(20) $$p(x) = \frac{d(x)}{n},$$

where n is the total number of neurons in neural population N; and $p(x)$ is the prior over stimulus values. Accordingly, they propose a model on which the prior is encoded by the density function via (20). In the model, there is no need for a separate population that encodes the prior. Instead, the prior is encoded by the allocation of resources across the population N whose stochastic behavior encodes the prior likelihood. See Figure 10. Note that the proposed decoder (20) is nonparametric: it maps the density function $d(x)$ to a unique prior $p(x)$, without any restriction as to parametric form.

INSERT FIGURE 10 ABOUT HERE

Ganguli and Simoncelli (2014, pp. 2117-2118) show that their decoder supports approximate computation of the posterior. The posterior is approximated by a discrete distribution:

(21) $$p_{new}(x_i) = \frac{e^{\sum_{j=1}^{n} r_j \log f(i-j)}}{\sum_{k=1}^{n} \left(e^{\sum_{j=1}^{n} r_j \log f(k-j)} \right)}.$$

By (19), $f(i-j)$ is proportional to neuron j's average response to neuron i's preferred stimulus value. (21) uses logarithms of these average responses to form a weighted sum of spike counts,

then exponentiates and normalizes. A neural network that executes the mandated operations can instantiate firing rate profile ***t*** over a separate neural population, where

(22) $$t_i = \frac{e^{\sum_{j=1}^{n} r_j \log f(i-j)}}{\sum_{k=1}^{n}\left(e^{\sum_{j=1}^{n} r_j \log f(k-j)}\right)}.$$

Firing rate profile ***t*** encodes p_{new} via the decoder

(23) $p_{new}(x_i) = t_i$.

Thus, a neural network that employs the density-based encoding scheme (20) can approximate the posterior.

Computational neuroscientists have proposed several other PPC implementation mechanisms (Beck, Heller, and Pouget, 2012; Orhan and Ma, 2017; Pouget et al., 2013). No doubt further PPC models will emerge in the near future.

§3.2 Sampling

I now discuss an alternative neural implementation strategy centered on sampling.

An early example is the *Boltzmann machine* (Ackley, Hinton, and Sejnowski, 1985). A Boltzmann machine consists of the following elements: a collection of n neuron-like units that can turn on and off ($z_i = 1$ means that unit i is on, $z_i = 0$ means that it is off); weights w_{ij}, codifying the connection strength between units i and j, such that $w_{ij} = w_{ji}$ and $w_{ii} = 0$; and bias terms b_i, codifying the propensity of unit i to take value 1. We may construe z_i as the neural network's current vote regarding the true value of some binary random variable X_i (e.g. whether a perceived object is concave or convex). The weights and bias terms encode a discrete probability distribution over the random variables $X_1,\ldots, X_n$ via the decoder

(24) $p(x_1,...,x_n) = \eta e^{H(x_1,...,x_n)}$,

where

$$H(x_1,...,x_n) = \sum_{i<j} w_{ij} x_i x_j + \sum_i b_i x_i \ ,$$

and η is a normalization constant.

Suppose that the variables $X_1,\ldots, X_n$ fall into two categories: observable ($X_1,\ldots, X_k$) and unobservable ($X_{k+1},\ldots, X_n$). We wish to form a new credence over the unobservable variables given observed values $x_1,\ldots, x_k$ of the observable variables. The posterior is

(25) $p(x_{k+1},\ldots, x_n \mid x_1,\ldots, x_k)$.

In principle, (25) can be computed directly from (24) using the ratio formula for conditional probabilities:

$$p(a \mid b) = \frac{p(a,b)}{p(b)} .$$

However, the computation is not typically tractable. We may instead approximate the posterior as follows. First, "clamp" units 1 through k to the values

$$z_1 = x_1,\ z_2 = x_2,\ \ldots,\ z_k = x_k.$$

Second, assign arbitrary values to the remaining units. Third, sample a new value z_i from unit $i > k$ according to the conditional chance distribution

(26) $$c(z_i = 1 \mid \boldsymbol{z}_{\backslash i}) = \frac{1}{1 + e^{-b_i - \sum_{j \neq i} w_{ij} z_j}} .$$

where $\boldsymbol{z}_{\backslash i}$ is the profile of values currently assigned to all the units besides unit i. Cycle through all the remaining units in the same way, holding fixed the clamped units. Continue in this way for some time, sampling values for the non-clamped units according to (26). At each stage, there is an objective chance $c(z_{k+1},\ldots, z_n)$ of sampling values $z_{k+1},\ldots, z_n$ from units $k + 1$ through n.

$c(z_{k+1},\ldots, z_n)$ will change as we continue to draw samples. One can show that $c(z_{k+1},\ldots, z_n)$ converges to the posterior $p(x_{k+1},\ldots, x_n \mid x_1,\ldots, x_k)$. If we run the sampling procedure for a sufficient "burn in" period and subsequently set

(27) $p_{new}(x_{k+1},\ldots, x_n) = c(z_{k+1},\ldots, z_n)$,

then $p_{new}(x_{k+1},\ldots, x_n)$ approximates the posterior $p(x_{k+1},\ldots, x_n \mid x_1,\ldots, x_k)$. This is an example of a sampling procedure known as *Gibbs sampling*, which itself is a special case of a more general sampling strategy known as *Metropolis-Hastings* (Murphy, 2012, pp. 839-876). See (Icard, 2016) for extended discussion of the Boltzmann machine and sampling propensities.

The Boltzmann machine is not very realistic from a neurophysiological perspective. It does not even model the basic fact that neurons emit spikes. Still, it nicely illustrates how neural networks can implement approximate Bayesian inference through sampling. Similar sampling implementations are achievable by far more biologically realistic neural networks.

An example is the neural network given by Buesing et al. (2011), which models in a biologically plausible way the stochastic interactions within a collection of n spiking neurons. $z_i = 1$ at time t signifies that neuron i has fired in a small time interval ending at t. $z_i = 0$ signifies that neuron i has not fired in that time interval. The weights and bias terms encode a probability distribution over n binary random variables $X_1,\ldots, X_n$ through the decoder (24).[5] Neuron i has membrane potential u_i, which is related to spiking activity by the equation:

$$u_i = b_i + \sum_{j \neq i} w_{ij} z_j \ ,$$

where the bias term b_i codifies neuron i's excitability; w_{ij} is the connection strength between neurons i and j; and z_i reflects the current spiking behavior (or lack thereof) of neuron i. To approximate the posterior

[5] The model can be generalized to handle other decoders (Buesing et al., 2011, p. 4). See also (Pecevski et al., 2011) for further generalizations.

$$p(x_{k+1},\ldots, x_n \mid x_1,\ldots, x_k),$$

the network proceeds in roughly the same fashion as the Boltzmann machine: it clamps the values of $z_1 = x_1$, $z_2 = x_2$, …, $z_k = x_k$, then serially samples values z_i of the remaining neurons. Samples are drawn stochastically, in a way that depends upon membrane potentials along with other neurophysiological details. Buesing et al. (2011) prove that their stochastic sampling procedure converges to the posterior $p(x_{k+1},\ldots, x_n \mid x_1,\ldots, x_k)$. Just as with the Boltzmann machine, we may run the sampling procedure for a "burn in" period and then set

$$p_{new}(x_{k+1},\ldots, x_n) = c(z_{k+1},\ldots, z_n).$$

The new credal state p_{new} approximates the true posterior.

One disadvantage shared by the Boltzmann machine and the (Buesing et al., 2011) model is that they only encode probability distributions over binary random variables. A model given by Nessler et al. (2013) uses sampling to compute the posterior over a discrete random variable X that takes k possible values x_1, …, x_k. The neural network contains n input neurons, whose spiking behavior is modeled by n binary random variables Y_1, …, Y_n. These neurons can code values of non-binary discrete sensory variables as long as n is large enough (where each input neuron corresponds to a distinct value of some sensory variable). Sample values of X are encoded by a population of k output neurons: a spike by output neuron i codes a sample x_i. Output neuron i's spiking propensity is determined by its membrane potential u_i. Membrane potential in turn depends upon input neuron spikes as follows:

$$u_i = b_i - I + \sum_{j=1}^{n} w_{ij} y_j \ ,$$

where the bias term b_i codifies output neuron i's excitability; w_{ij} is the connection strength between output neuron i and input neuron j; and I is an inhibition signal. The prior $p(x_i)$ is encoded by the neuron excitability profile via the decoder:

(28) $p(x_i) = e^{b_i}$.

The prior likelihood is encoded by the weights w_{ij} via the decoder

(29) $p(\boldsymbol{y} \mid x_i) = e^{\sum_{j=1}^{n} w_{ij} y_j}$,

where $\boldsymbol{y} = \langle y_1, y_2, \ldots, y_n \rangle$. Here we must assume that the bias terms and weights meet normalization conditions, so that (28) and (29) yield normalized probabilities. Nessler et al. (2013) show that, under some additional assumptions, spiking propensities among the output neurons match the posterior $p(x_i \mid \boldsymbol{y})$. If $p_{new}(x_i)$ is encoded by the chances governing output neuron spikes, then $p_{new}(x_i)$ is simply the posterior $p(x_i \mid \boldsymbol{y})$.[6]

Most variables encountered in perception are continuous (e.g. shape, size, color, location) rather than discrete. The literature offers several neural network models that sample from the (approximate) posterior for a continuous random variable (e.g. Aitchison and Lengyel, 2016; Hennequin et al., 2014; Moreno-Bote et al., 2011; Savin and Denève, 2014). The basic idea is usually that samples are encoded by values of a continuous neural variable, such as a neuron's membrane potential (Orbán et al., 2016). The objective chance function governing this neural variable encodes the (approximate) posterior. For example, the decoder might take the form:

(30) $p_{new}(x) = c(f(x))$,

where $f(x)$ is the membrane potential that encodes stimulus value x and c is an objective chance function governing membrane potentials. Sampling neural networks are under active investigation, so we may expect the coming years to bring forth additional models.[7]

[6] See also the sampling-based neural network given by Huang and Rao (2016), which models iterated approximate Bayesian inference for arbitrary probability distributions over a finite space.

[7] Some particle filter neural implementation models feature an encoding scheme along the lines of note 2. A good example is the model given by Kutschireiter et al. (2017), which implements iterated approximate Bayesian inference for finitely many continuous random variables.

§3.3 Predictive coding

Recent philosophical literature places great emphasis on *predictive coding* (Clark, 2015; Hohwy, 2014). The basic idea is that the neural network generates a prediction α about sensory input. Upon receipt of actual sensory input y, the network computes a *prediction error* term. Typically (e.g. Rao and Ballard, 1999), prediction error ε is the difference

$$\varepsilon = y - \alpha.$$

Alternatively (e.g. Spratling, 2016), prediction error may be the quotient

$$\varepsilon = y/\alpha.$$

Either way, prediction error figures prominently in subsequent computation. For example, it may influence future predictions. Many predictive coding models have hierarchical structure: higher levels of the network pass predictions down to lower levels, and lower levels pass prediction errors back to higher levels.[8]

There is nothing inherently Bayesian about predictive coding (Aitchison and Lengyel, 2017). However, if one sets up the neural network in the right way, then predictive coding can implement approximate Bayesian inference. Consider the hierarchical neural network given by Spratling (2016). Each level of the hierarchy contains three neural populations: the first computes a vector $\boldsymbol{\alpha}$ of sensory input predictions; the second combines $\boldsymbol{\alpha}$ with sensory input $\boldsymbol{y}$ to compute prediction error $\boldsymbol{\varepsilon}$; the third uses $\boldsymbol{\varepsilon}$ to update the estimate $\boldsymbol{x}$ of the underlying distal variable. The update of $\boldsymbol{x}$, which depends upon a matrix $\boldsymbol{W}$ of feedforward weights, is used to update sensory prediction $\boldsymbol{\alpha}$. The prior probability is encoded as a scaling factor that modulates the feedforward weight matrix $\boldsymbol{W}$. The prior likelihood is encoded by a population of input neurons with independent Poisson variability. The posterior is encoded by firing rates in the

[8] See (Cao, 2020) for critical discussion of talk about "prediction" and "prediction error" in this context.

prediction neurons, where each prediction neuron has a preferred stimulus value. Under this decoding scheme, Spratling shows that the network can compute an approximate posterior for Gaussian priors and some non-Gaussian priors.

The literature offers various alternative predictive coding implementations of approximate Bayesian inference. For example, Lee and Mumford (2003) offer a sampling-based predictive coding implementation of iterated approximate Bayesian inference, while Friston (2005, 2010) develops a predictive coding implementation that computes a variational approximation to the posterior. See (Spratling, 2017) for an overview.

§4. Morals

The previous section canvassed several theories of how the brain implements approximate Bayesian inference. I will not consider neurophysiological evidence for or against the theories. Instead, I want to advance five morals that we can draw quite apart from which theory (if any) turns out to be correct.

***First moral*: *There are diverse biologically plausible candidate neural realizers for credal states*.** A neural realizer for the prior probability is a neural variable U that, at a bare minimum, satisfies appropriate counterfactuals of the form (4). A neural realizer for the prior likelihood is a neural variable V that, at a bare minimum, satisfies appropriate counterfactuals of the form (6).[9] A neural realizer for the approximate posterior is a neural variable W that, at a bare minimum, satisfies appropriate counterfactuals of the form (8). §3 canvassed several candidate neural realizers:

[9] In some models, such as (Ma et al., 2006) and (Ganguli and Simoncelli, 2014), the neural network executes an approximate Bayesian inference based on spike count profile $\boldsymbol{r}$ over a neural population. $\boldsymbol{r}$ is caused by proximal sensory input e. However, e does not enter directly into the inference. Accordingly, the neural network realizes a prior likelihood $p(\boldsymbol{r} \mid x)$ defined over spike count $\boldsymbol{r}$ rather than proximal sensory input e. The rationale here is that neural computation has direct access to $\boldsymbol{r}$ rather than e.

- firing rate profile over a neural population: equations (10), (11), and (23)
- spike count profile over a neural population: equations (16) and (18)
- chance distribution governing neural response to a stimulus: equation (14)
- tuning curve density function: equation (20)
- sampling propensities: equations (27) and (30)
- neuron excitability profile: equation (28)
- weights in a neural network: equations (24) and (29)

These candidates are all under scientific active investigation, as are other candidates. The candidates range from the relatively concrete (e.g. spike count profile) to the highly abstract (e.g. tuning curve density function).

Second moral*: *Credal assignments may be implicit. None of the models we have considered feature explicit enumeration of prior probabilities. The closest is equation (10), where r_i may be construed as encoding the probability assigned to x_i. But even (10) does not feature true explicit enumeration: first, firing rates are normalized to yield probabilities; second, the encoding scheme implicitly specifies that stimulus values other than preferred values x_i receive probability 0. In the other encoding schemes, credal assignment are even more implicit. An extreme example of implicit encoding is the tuning curve density function $d(x)$. *We theorists* represent $d(x)$, but *the neural network itself* does not represent $d(x)$. Assuming decoder (20), the prior is not explicitly recorded anywhere in the network's computations. Instead, it is implicitly enshrined by the neural network's allocation of resources.

***Third moral: The prior probability and the posterior may have very different neural realizers*.** In (Ganguli and Simoncelli, 2014), the prior probability is realized by tuning curve density $d(x)$ via equation (20), while the posterior is encoded by firing rate profile via equation

(23). In (Nessler et al., 2013), the prior probability is encoded by the neuron excitability profile via equation (28), while the posterior is encoded by sampling propensities. These examples demonstrate that a single neural network may realize credal assignments in different ways at different stages of computation. The examples vividly illustrate *multiply realizability*, a crucial mark of the mental first highlighted by Putnam (1967). A psychological state type (in this case, a credal assignment over a hypothesis space) may have distinct tokens that are quite diverse at the neural level. Our examples show that distinct tokens may be neurally diverse *even within a single biologically plausible neural system*.

Fourth moral: The prior probability and the prior likelihood may or may not be separately encoded. They are separately encoded in most of the models I considered. In (Ma et al., 2006), for example, the prior probability is encoded by spike counts in a neural population via equation (16), while the prior likelihood is encoded by objective chances via equation (14). In some models, though, a single neural state encodes both the prior probability *and* the prior likelihood. The Boltzmann machine encodes a prior $p(x_1,\ldots, x_n)$ via equation (24), and the encoded prior determines all relevant unconditional and conditional probabilities ---- including the prior likelihood $p(x_1,\ldots, x_k \mid x_{k+1},\ldots, x_n)$. Similarly for the (Buesing et al., 2011) model. Nothing about the Bayesian framework requires separate encoding of the prior $p(h)$ and the prior likelihood $p(e \mid h)$. One can instead encode a prior $p(e, h)$ and then define conditional probabilities via the ratio formula. In that case, the prior probability and the prior likelihood are encoded by the same neural variable. In terms of clauses (5) and (7): $U = V$, and Φ maps u to the prior probability $p(h)$ while Ψ maps v ($= u$) to the prior likelihood $p(e \mid h)$.

Fifth moral: There are diverse biologically plausible candidate neural implementation mechanisms for approximate Bayesian inference. Physical implementation of approximate

Bayesian inference can be achieved, at least in principle, through diverse neural operations falling squarely within the repertoire of the human brain. In (Ma et al., 2006), approximate Bayesian inference is implemented by linear combination of spike counts. In (Ganguli and Simoncelli, 2014), it is implemented by linear combination of spike counts, exponentiation, and normalization. In sampling models, it is implemented by a sampling algorithm. In (Spratling, 2016), it is implemented by computation of prediction errors. These implementation schemes are under active scientific investigation, with various pieces of empirical evidence for or against each candidate scheme. In particular, predictive coding is just one proposed neural implementation among others. Many proposed implementations do not involve anything like computation of prediction error. At least some of those proposed implementations have just as much empirical support as any known predictive coding implementation (Aitchison and Lengyel, 2017).

§5. Simulation, not implementation

I have been discussing the diverse ways that a neural network might implement approximately Bayesian inference. I will now discuss neural networks that *simulate* rather than *implement* approximate Bayesian inference.

In a typical Bayesian perceptual model, the new credal state $p_{new}(h)$ is not the final output but instead is used to select a privileged estimate h^*. The model determines a deterministic or stochastic mapping from inputs e to estimates h^*. In principle, there are several ways a neural network might instantiate the desired mapping from e to h^* *without* implementing the credal transition depicted by Figures 4 and 5:

- *No priors, no approximate posterior* (Figure 11). As noted in §2, the mapping from e to h^* could be implemented by a machine that consults a look-up table. Alternatively, we can sometimes train a neural network to implement the mapping (Simoncelli, 2009). There are even circumstances where *unsupervised* learning enables a system to emulate a Bayesian estimator (Raphan and Simoncelli, 2007). Thus, a neural network can mimic Bayesian estimation without instantiating any credal states or transitions.
- *Prior likelihood, approximate posterior, but no prior probability* (Figure 12). A system may encode the prior likelihood and an approximate posterior but not the prior probability. To illustrate, consider a simplified version of the (Ma et al., 2006) model. As we have seen, the prior likelihood $p(\boldsymbol{r} \mid x)$ is encoded by the objective chance distribution governing a neural population N, via equation (14). Assuming a flat prior, one can show that the posterior $p(x \mid \boldsymbol{r})$ is a Gaussian with parameters determined by (15). So, assuming a flat prior, there is no need for separate encoding of the prior or separate computation of the posterior: the spike count profile $\boldsymbol{r}$ over N itself already encodes the posterior. See (Ma et al., 2006, pp. 1433-1444) for discussion.
- *Prior probability, approximate posterior, but no prior likelihood* (Figure 13). A system may encode the prior probability and the approximate posterior but not the prior likelihood. Consider the model given by Rullán Buxó and Savin (2021), which combines sampling and parametric encoding. Firing rates in neural population N_1 encode samples from a probability distribution over random variables $X_1,\ldots, X_n$. At first, N_1 samples spontaneously from the prior $p(x_1,\ldots, x_n)$. Upon receiving sensory input e, N_1 samples from the posterior $p(x_1,\ldots, x_n \mid e)$. Samples produced by N_1 serve as input to a second neural population N_2, which computes parameters for an

approximate posterior $p_{new}(x_i)$ over a single variable of interest X_i. Although the prior probability and the approximate posterior are implicitly encoded, the prior likelihood $p(e \mid x_1,\ldots, x_n)$ is not. No neural variable realizes $p(e \mid x_1,\ldots, x_n)$. Instead, the prior likelihood is embedded in the sampling dynamics for N_1.

- *Priors, but no approximate posterior* (Figure 14). A system may encode the prior probability and the prior likelihood but transform input e into estimate h^* without computing an approximate posterior. Consider the predictive coding model given by Rao and Ballard (1999). The neural network encodes the prior probability and the prior probability. In response to input e, the network uses a predictive coding algorithm to select the MAP estimate. As Rao (2004, pp. 29-30) notes, the network does not compute $p(h \mid e)$ or any approximation to $p(h \mid e)$. It only computes argmax_h $p(h \mid e)$. So it does not implement a credal transition (a transition among credal states).
- *Approximate posterior, but no priors* (Figure 15). We might train a system to compute $p(h \mid e)$ in response to input e even though the system does not encode the prior probability or the prior likelihood. An example is the neural network given by Echeveste et al. (2020), which was trained to respond to input e with a sampling-based encoding of the posterior $p(h \mid e)$.

A neural network that implements the mapping from e to h^* need not instantiate Figure 5. It might instead instantiate one of Figures 11-15.

INSERT FIGURES 11-15 ABOUT HERE

A key feature that differentiates Figure 5 from Figures 11-15 is the presence of neural realizers for credal states. Neural networks conforming to Figure 5 feature neural realizers for the prior probability, the prior likelihood, and the approximate posterior. Figures 11-15 depict situations where at least one of those three credal states lacks a neural realizer. Genuine implementation of approximate Bayesian inference requires that all three credal states have neural realizers.

If a neural network maps inputs e to estimates h^* in accord with an approximately Bayesian model, then the network's activity must reflect the model's priors *in some way*. The question is *how* it reflects them. According to Figure 5, priors are encoded by states of the neural network. The network applies general neural operations (e.g. linear combination of spike counts), yielding a new neural state that encodes a new credal state $p_{new}(h)$. Figures 11-15 diverge from that picture to varying degrees.

Take Figure 12. Here the prior probability is not encoded by any neural state but is instead subsumed into the network dynamics: the transition from the prior likelihood to the approximate posterior is only appropriate assuming the fixed prior probability. Depending on the details, it may be difficult or impossible to change the network dynamics to reflect a different prior. Figure 5 posits a flexible dynamics that can accommodate different priors, while Figure 12 posits a dynamics tailored to a specific prior. Figure 5 views the prior as an adjustable parameter that can change even as the network dynamics remains fixed. Figure 12 recognizes no such adjustable parameter. To illustrate, compare the (Ma et al., 2006) model in two versions: the version from §3.1, with a separate neural population N_2 that encodes the prior; and the version captured by Figure 12, in which the dynamics is tailored to a flat prior. In the first version, the prior is an adjustable parameter. We can change it (by changing spike counts in N_2) without

changing the network dynamics. In the second version, the flat prior is not an adjustable parameter. Incorporating a non-flat prior would require radical changes to the network dynamics.

A similar contrast applies to Figure 5 versus Figure 13: the former posits a flexible dynamics that can accommodate different prior likelihoods, while the latter posits a dynamics tailored to a specific prior likelihood. The contrast is even starker for Figures 11 and 15, which posit a dynamics tailored to a specific prior probability *and* a specific prior likelihood.

Now compare Figure 5 with Figure 14. Figure 5 computes the full approximate posterior, while Figure 14 only computes an estimate h^*. Networks conforming to Figure 5 can, at least in principle, support computations that networks conforming to Figure 14 cannot. A neural network that encodes the approximate posterior can in principle execute (or be supplemented so as to execute) the following computations:

- *expected value computation* relative to the approximate posterior.
- *probability matching*, i.e. stochastically selecting a privileged estimate h^* with objective chance given approximately by the approximate posterior.
- *further approximate Bayesian inference*, with the approximate posterior serving as the new prior.

These computations may not always be possible in practice. But the implementation schemes surveyed in §§3-4 support at least some of the computations through biologically plausible neural operations. For example, the Ganguli and Simoncelli (2014) model supports expected value computation, and sampling models trivially support probability matching. In contrast, a neural network that does not encode the approximate posterior cannot execute *any* such computations. Crucial information is irretrievably lost when a network encodes only a privileged

estimate h^* rather than an approximate posterior. Thus, the contrast between Figures 5 and 14 has significant implications for future computation.

Our discussion highlights a crucial advantage offered by neural networks that implement approximate Bayesian inference versus neural networks that merely simulate approximate Bayesian inference: *flexibility*. Neural realization of the priors enables a flexible dynamics that can remain fixed as the priors change. It thereby supports Bayesian estimation across changing environmental conditions. Neural realization of the approximate posterior enables flexibility regarding which future computations can be executed. It thereby supports more computational options than are supported by selection of a privileged estimate h^*. Hence, implementation offers greater computational flexibility than mere simulation.[10]

§6. Methodological implications of realism

The debate between realism and instrumentalism has major methodological implications for computational neuroscience. From the realist perspective, we should expect to find neural realizers for the credal states posited by empirically successful Bayesian perceptual models. We should take Figure 5 as a guide to underlying neural activity. From the instrumentalist perspective, there is no reason to take Figure 5 as a guide. There is no reason to expect that we will find neural realizers for priors or the approximate posterior.

Figures 11-15 decline to extend the realist viewpoint towards the prior probability, the prior likelihood, or the approximate posterior. Importantly, though, only Figure 11 embodies

[10] In the machine learning literature, it is standard to distinguish between *generative* versus *discriminative* models (Murphy, 2012, pp. 270-279). Basically, a generative model uses the prior $p(h)$ and the prior likelihood $p(e \mid h)$ to compute the posterior $p(h \mid e)$, while a discriminative model computes the posterior or some function of the posterior but does not encode the priors. Generative models correspond to Figure 5. Discriminative models correspond either to Figure 11 or to Figure 15, depending on whether the model merely maps e to a function of $p(h \mid e)$ or whether the model computes $p(h \mid e)$ itself. Increased computational flexibility is widely recognized as an advantage offered by generative models over discriminative models (Murphy, 2012, p. 271).

total rejection of realism. Each other figure extends the realist viewpoint towards either the prior probability, the prior likelihood, or the approximate posterior. So each other figure embodies what one might call *local realism* regarding specific elements of Bayesian models (e.g. realism regarding the prior probability but not the prior likelihood or the approximate posterior). Local realism conflicts with instrumentalism, which declines to extend the realist viewpoint towards any credal states posited by Bayesian models.[11]

My analysis hinges upon neural realization of credal states, yet I have not said what makes it the case that a neural variable realizes a credal state. To illustrate, assume for the sake of argument that spike count profile realizes the prior according to decoder (16). Why that particular decoder rather than some other decoder or no decoder at all? Lacking concrete answers to such questions, some readers may feel that I have not identified any substantive difference between implementing versus merely simulating approximate Bayesian inference. It might seem that one can always read the causal structure from Figure 5 into a physical system that emulates approximate Bayesian inference. One need merely isolate variables U, V, and W that mediate in the appropriate way between e and h^*. One can then interpret those variables using whatever

[11] Sohn and Narain (2021) distinguish two perspectives on neural implementation of Bayesian inference: the *modular* perspective and the *transform* perspective. According to the modular perspective, "probabilistic computations are carried out using independent representations of likelihood, prior, and posterior distributions, followed by the generation of an estimate" (p. 123). They cite (Ma et al., 2006) as an exemplar of the modular perspective. According to the transform perspective, "uncertain sensory measurements can be directly mapped into Bayesian estimates via latent processes within which prior distributions are embedded. This process does not mandate encoding of probabilistic distributions on each trial" (pp. 122-123). The transform perspective does not require encoding of priors or prior likelihoods, nor does it require computation of the posterior. It only requires that the system emulate Bayesian estimation. Thus, it encompasses Figure 11-15. Sohn and Narain (p. 124) also cite the (Ganguli and Simoncelli, 2014) model as an example of the transform perspective rather than the modular perspective. In that model, the prior and the likelihood functions are not *independently* encoded: the density $d(x)$ warps the tuning curves $f_i(x)$ and thereby influences the prior likelihood (14); so a change in the prior will generally induce a change in the likelihoods. I believe that the contrast between the modular and transform perspectives, while useful for some purposes, blurs vital distinctions. There is a significant difference between the (Ganguli and Simoncelli, 2014) model and a neural network with the causal structure of Figure 11: the former uses an implicitly encoded prior and prior likelihood to compute the posterior; the latter does not. Even though the (Ganguli and Simoncelli, 2014) model does not encode the prior and the prior likelihood *independently*, it seems closer in many important respects to the (Ma et al., 2006) model than to a neural network that merely emulates Bayesian estimation.

decoders Φ, Ψ, and Γ one pleases, thereby depicting the system as conforming to Figure 5. Apparently, the contrast I have drawn between realism and instrumentalism evaporates.[12]

I find this line of thought misplaced for several reasons. First, it is hardly obvious that we can find suitable neural variables *U*, *V*, and *W* that mediate between *e* and *h**. We should allow only variables that a neuroscientist would take seriously --- e.g., spike count, firing rate, synaptic weight, etc. We should disallow disjunctive or gerrymandered variables. This restriction severely limits our ability to read the causal structure from Figure 5 into a physical system. Second, one cannot simply interpret a neural variable using whatever decoder one pleases. Whether a neural state realizes a credal state is not a matter of interpretation. Either the neural state realizes the credal state or it does not. Admittedly, I have not given necessary and sufficient conditions for realizing a credal state. But this does not mean that anything goes. As mentioned in §3, a neural state can realize a prior over a distal variable only if the neural state bears appropriate causal connections to the distal variable. Quite plausibly, the neural state must also figure or potentially figure in some characteristic Bayesian computations, such as computation of expected value or of an approximate posterior. More generally, a neural state realizes a credal state only if the neural state is appropriately related to the distal environment and to other neural states. Given these restrictions on variables and decoders, it is not so easy to read Figure 5 into any arbitrary system that emulates approximate Bayesian estimation. For example, there is no evident way to impose Figure 5 upon a system that emulates Bayesian estimation using a look-up table. To take a more realistic example, there is no evident way to depict the (Rao and Ballard, 1999) predictive coding model as including a neural realizer for the approximate posterior.

---

[12] This worry is closely connected *triviality arguments regarding computational implementation*, propounded by Putnam (1988) and Searle (1990). For critical discussion of triviality arguments, see (Rescorla, 2013; 2014a).

Obviously, we would like to clarify physical realization of credal states. The key point for present purposes is that, even lacking the desired clarification, realism entrains fundamentally different methodological commitments than instrumentalism. If we adopt a realist viewpoint towards a credal state, then we should seek a neural realizer for the credal state. If we extend the realist viewpoint towards the prior probability, the prior likelihood, and the posterior, then we should seek neural realizers for all three credal states. Instrumentalists see no need to seek neural realizers for any credal states. In practice, then, realists and instrumentalists will tend to pursue very different models of neural computation.

Should we adopt a realist viewpoint towards the credal states posited within Bayesian perceptual psychology? In my opinion, there is strong evidence that perceptual computation exhibits the flexibility characteristic of Figure 5:

- *Change in the prior probability*. To illustrate, consider a well-known perceptual illusion: when a moving line is viewed at low contrast, its perceived direction of motion is biased towards the perpendicular. The (Weiss, Simoncelli, and Adelson, 2002) motion estimation model explains this illusion through the "slow motion" prior: the illusory perpendicular velocity is slower than the true velocity. Sotiropoulos, Seitz, and Seriès (2011) exposed subjects to fast moving parallel lines. After exposure, subjects tended to perceive the lines as moving obliquely rather than perpendicularly, corresponding to a faster speed. The change in motion perception is well-explained by a shift in the "slow motion" prior to favor faster speeds.[13]
- *Change in the prior likelihood*. In many cases, sensory adaptation is well-explained by a change in the prior likelihood. Consider the *ventriloquism illusion*: if there is a

[13] Another example of flexibility: new priors can *transfer* from one perceptual task to another (Adams, Graf and Ernst, 2001; Maloney and Mamassian, 2009). See (Rescorla, 2020b) for discussion in support of realism.

conflict between visual and auditory cues to stimulus location, the visual system heavily favors the visual cue when forming a unified location estimate. The ventriloquism illusion can be explained in Bayesian terms, as an inference based on the visual cue and the auditory cue (Alais and Burr, 2004). Repeated exposure to the ventriloquism illusion induces the *ventriloquism aftereffect*, in which location estimates based solely on the auditory cue are systematically altered. Sato, Toyoizumi, and Aihara (2007) show that the ventriloquism aftereffect is well-explained by a shift in the prior likelihood relating location estimates to auditory cues. Intuitively: sustained exposure to ventriloquism changes the auditory stimulation that the perceptual system expects from a given stimulus location.

- *Computations that exploit the (approximate) posterior*. There is strong evidence that the perceptual system can sometimes execute computations exploiting the approximate posterior (Koblinger, Fiser, and Lengyel, 2021). A good example is the object-tracking model given by Kwon, Tadin, and Knill (2015). The model posits sequential Bayesian estimation of position and velocity in response to sequential sensory input. At each stage, the Bayesian estimator executes a new probabilistic inference, taking the posterior from the previous stage as the new prior. The model explains a range of motion illusions that otherwise resist unified explanation.

These are just some representative examples. Overall, the scientific literature offers strong psychophysical evidence for flexible perceptual computations that fit better with a realist approach to credal states than with an instrumentalist approach (Rescorla, 2020b).

Instrumentalists may hope to explain the psychophysical evidence through alternative anti-realist explanations. In that spirit, Block (2018) suggests that one might explain apparent

changes in the prior probability through a model that simulates Bayesian estimation and *also* simulates a changing prior caused by changing environmental conditions. However, it is not enough merely to suggest that some possible theory might explain the change in direction perception documented by Sotiropoulos, Seitz, and Seriès (2011). One must propose an actual theory that explains the observed phenomena without positing a changed prior. One must then compare the proposed theory with the realist alternative. So far, this has not happened. Instrumentalists have not proposed alternative explanations that abjure credal states and transitions, let alone argued that such explanations can equal or surpass explanations that posit credal states and transitions.

I think that we currently have good reason to favor realism over instrumentalism. We have good reason to take Figure 5 as a guide to neural activity. As I have documented, lots of research within computational neuroscience pursues precisely that realist agenda. The agenda has proved fruitful, with several recent studies supplying suggestive neurophysiological evidence for neural realization of credal states (Berkes et al., 2011; Sohn and Narain, 2021; Walker et al., 2020). Future scientific developments will reveal whether the realist agenda yields well-confirmed models of the brain.

Acknowledgments

I thank Rosa Cao, Thomas Icard, Jiarui Qu, Susanna Schellenberg, Nicholas Shea, and the editors for helpful feedback on an earlier draft of this paper. I am also grateful to Jiarui Qu for preparing Figures 1-15.

Works Cited

Ackley, D., Hinton, G., and Sejnowski, T. 1985. "A Learning Algorithm for Boltzmann Machines." *Cognitive Science* 9: pp. 147-169.

Adams, W., Graf, E., and Ernst, M. 2004. "Experience Can Change the "Light-from-Above" Prior. *Nature Neuroscience* 7: pp. 1057-1058.

Aitchison, L., and Lengyel, M. 2016. "The Hamiltonian Brain: Efficient Probabilistic Inference with Excitatory-Inhibitory Neural Circuit Dynamics." *PLoS Computational Biology* 12: e1005186.

Aitchison, L., and Lengyel, M. 2017. "With or Without You: Predictive Coding and Bayesian Inference in the Brain." *Current Opinion in Neurobiology* 46: pp. 219-227.

Alais, D., and Burr, D. 2004. "The Ventriloquist Effect Results from Near-optimal Bimodal Integration." *Current Biology* 14: pp. 257-262.

Barber, M., Clark, J., Anderson, C. 2003. "Generating Neural Circuits that Implement Probabilistic Reasoning." *Physical Review E* 68: 041912.

Beck, J., Ma, W. J., Latham, P. E., and Pouget, A. 2007. "Probabilistic Population Codes and the Exponential Family of Distributions." In *Computational Neuroscience: Theoretical Insights into Brain Function*, eds. P. Cisek, T. Drew, and J. F. Kalaska. New York: Elsevier.

Beck, J., Heller, K. and Pouget, A. 2012. "Complex Inference in Neural Circuits with Probabilistic Population Codes and Topic Models." In *Advances in Neural Information Processing Systems*, ed. P. Bartlett. Cambridge: MIT Press.

Bennett, K. 2007. "Mental Causation." *Philosophy Compass* 2: pp. 316-337.

Berkes, P., Orbán, G., Lengyel, M., and Fiser, J. 2011. "Spontaneous Cortical Activity Reveals Hallmarks of an Internal Model of the Environment." *Science* 331: pp. 83-7.

Block, N. 2018. "If Perception is Probabilistic, Why Does It Not Seem Probabilistic?". *Philosophical Transactions of the Royal Society B* 373: 20170341.

Buesing, L., Bill, J., Nessler, B., and Maass, W. 2011. "Neural Dynamics as Sampling: A Model for Stochastic Computation in Recurrent Networks of Spiking Neurons." *PloS Computational Biology* 7: e1002211.

Burge, T. 2007. *Foundations of Mind*. Oxford: Clarendon Press.

---. 2010. *Origins of Objectivity*. Oxford: Oxford University Press.

Cao, R. 2020. "New Labels for Old Ideas: Predictive Processing and the Interpretation of Neural Signals." *Review of Philosophy and Psychology* 11: pp. 517-546.

Clark, A. 2015. *Surfing Uncertainty*. Oxford: Oxford University Press.

Colombo, M., and Seriès, P. 2012. "Bayes on the Brain --- on Bayesian Modeling in Neuroscience." *The British Journal for the Philosophy of Science* 63: pp. 697-723.

Crisan, D., and Doucet, A. 2002. "A Survey of Convergence Results on Particle Filtering Methods for Practitioners." *IEEE Transaction on Signal Processing* 50: pp. 736-746.

Dayan, P., and Abbott, L. F. 2004. *Theoretical Neuroscience*. Cambridge: MIT Press.

Echeveste, R, Aitchison, L., Hennequin, G., and Lengyel, M. 2020. "Cortical-like Dynamics in Recurrent Circuits Optimized for Sampling-based Probabilistic Inference." *Nature Neuroscience* 23: pp. 1138-1149.

Echeveste, R., and Lengyel, M. 2018. "The Redemption of Noise: Inference with Neural Populations." *Trends in neurosciences* 41: pp. 767-770.

Egan, F. 2010. "Computational Models: A Modest Role for Content." *Studies in History and*

Philosophy of Science 41: pp. 253–259.
Erikkson, L., and Hájek, A. 2007. "What are Degrees of Belief?". *Studia Logica* 86: pp. 183-213.
Ernst, M. 2007. "Learning to Integrate Arbitrary Signals from Vision and Touch." *Journal of Vision* 7: pp. 1-14.
Fiser, J., Berkes, P., Orbán, G., and Lengyel, M. 2010. "Statistically Optimal Perception and Learning: From Behavior to Neural Representations." *Trends in Cognitive Science* 14: pp. 119–130.
Friston, K. 2005. "A Theory of Cortical Responses." *Philosophical Transactions of the Royal Society B* 360: pp. 815-836.
---. 2010. "The Free-energy Principle: A Unified Brain Theory?". *Nature Reviews Neuroscience* 11: pp. 127-138.
Ganguli, D., and Simoncelli, E. 2014. "Efficient Sensory Encoding and Bayesian Inference with Heterogeneous Neural Populations." *Neural Computation* 26: pp. 2103-2134.
Gelman, A., Carlin, J., Stern, H., Dunson, D., Vehatri, A., & Rubin, D. 2014. *Bayesian Data Analysis*, 3rd ed. New York: CRC Press.
Gershman, S., and Beck, J. 2017. "Complex Probabilistic Inference: From Cognition to Neural Computation." In *Computational Models of Brain and Behavior*, ed. A. Moustafa. Hoboken: Wiley-Blackwell.
Gershman, S., Vul, E., and Tenenbaum, J. 2012. "Multistability and Perceptual Inference." *Neural Computation* 24: pp. 1-24.
Hennequin, G., Aitchison, L., and Lengyel, M. 2014. "Fast Sampling-Based Inference in Balanced Neuronal Networks." In *Proceedings of the 27th International Conference on Neural Information Processing Systems* 2: pp. 2240-2248.
Hohwy, J. 2014. *The Predictive Mind*. Oxford: Oxford University Press.
Huang, Y., and Rao, R. 2016. "Bayesian Inference and Online Learning in Poisson Neuronal Networks." *Neural Computation* 28: pp. 1503-1526.
Icard, T. 2016. "Subjective Probability as Sampling Propensity." *The Review of Philosophy and Psychology* 7: pp. 863-903.
Knill, D., and Pouget, A. 2004. "The Bayesian Brain: The Role of Uncertainty in Neural Coding And Computation." *Trends in Neuroscience* 27: pp. 712–719.
Knill, D., and Richards, W. (eds.). 1996. *Perception as Bayesian Inference*. Cambridge: Cambridge University Press.
Koblinger, Á., Fiser., J., and Lengyel, M. 2021. "Representations of Uncertainty: Where Art Thou?". *Current Opinion in Behavioral Sciences* 38: pp. 150-162.
Kutschireiter, A., Surace, S. C., Sprekeler, H., and Pfister, J.-P. 2017. "Nonlinear Bayesian Filtering and Learning: A Neuronal Dynamics for Perception." *Scientific Reports* 7: 8722.
Kwisthout, J., Wareham, T., and van Rooij, I. 2011. "Bayesian Intractability is Not an Ailment that Approximation Can Cure." *Cognitive Science* 35: pp. 779-784.
Kwon, O.-S., Tadin, D., Knill, D. 2015. "Unifying Account of Visual Motion and Position Perception." *Proceedings of the National Academy of Sciences* 112: pp. 8142-8147.
Lee, T. S., and Mumford, D. 2003. "Hierarchical Bayesian Inference in the Visual Cortex." *Journal of the Optical Society of America* 20: pp. 1434-1448.
Loewer, B. 1997. "A Guide to Naturalizing Semantics." In *A Companion to the Philosophy of Language*, eds. C. Wright and B. Hale. Oxford: Blackwell.
Ma, W. J. 2019. "Bayesian Decision Models: A primer." *Neuron* 104: pp. 164-175.

Ma, W. J., Beck, J., Latham, P., and Pouget, A. 2006. "Bayesian Inference with Probabilistic Population Codes." *Nature Neuroscience* 9: pp. 1432-1438.
Maloney, L., and Mamassian, P. 2009. "Bayesian Decision Theory as a Model of Human Visual Perception: Testing Bayesian Transfer." *Visual Neuroscience* 26: pp. 147-155.
Moreno-Bote, R., Knill, D., and Pouget, A. 2011. "Bayesian Sampling in Visual Perception." *Proceedings of National Academy of Sciences* 108: 12491-6.
Murphy, K. 2012. *Machine Learning: A Probabilistic Perspective*. Cambridge: MIT Press.
Nessler, B., Pfeiffer, M., Buesing, L., and Maass, W. 2013. "Bayesian Computation Emerges in Generic Cortical Microcircuits through Spike-Timing-Dependent Plasticity." *PloS Computational Biology* 9: e1003037.
Orbán, G., Berkes, P., Fiser, J., and Lengyel, M. 2016. Neural variability and sampling-based probabilistic representations in the visual cortex. *Neuron* 92: pp. 530-542.
Orhan, A. E., and Ma, W. J. 2017. "Efficient Probabilistic Inference in Generic Neural Networks Trained with Non-probabilistic Feedback." *Nature Communications* 8: pp. 1-14.
Orlandi, N. (2014). *The Innocent Eye: Why Vision is not Cognitive Process*. Oxford: Oxford University Press.
Pecevski, D., Buesing, L., and Maass, W. 2011. "Probabilistic Inference in General Graphical Models through Sampling in Stochastic Networks of Spiking Neurons." *PloS Computational Biology* 7: e1002294.
Pouget, A., Dayan, P., and Zemel, R. 2003. "Inference and Computation with Population Codes." *Annual Review of Neuroscience* 26: pp. 381-410.
Pouget, A., Beck, J., Ma., W. J., & Latham, P. 2013. "Probabilistic Brains: Knowns and Unknowns." *Nature Neuroscience* 16: pp. 1170–1178.
Putnam, H. 1967. "Psychophysical Predicates." In *Art, Mind, and Religion*, eds. W. Capitan and D. Merrill. Pittsburgh: University of Pittsburgh Press.
---.1988. *Representation and Reality*. Cambridge: MIT Press.
Ramsey, F. P. 1931. "Truth and Probability." *The Foundations of Mathematics and Other Logical Essays*, ed. R. B. Braithwaite. London: Routledge and Kegan.
Rao, R. 2004. "Bayesian Computation in Recurrent Neural Circuit." *Neural Computation* 16: pp. 1-38.
Rao, R., and Ballard, D. 1999. "Predictive Coding in the Visual Cortex: A Functional Interpretation of Some Extra-classical Receptive-field Effects." *Nature Neuroscience* 2: pp. 79-87.
Raphan, M., and Simoncelli, E. 2007. "Learning to be Bayesian Without Supervision." In *Advances in Neural Information Processing Systems*, volume 19, eds. B. Schölkopf, J. Platt, and T. Hofmann. Cambridge: MIT Press.
Rescorla, M. 2013. "Against Structuralist Theories of Computational Implementation." *The British Journal for the Philosophy of Science* 64: pp. 681-704.
---. 2014a. "The Causal Relevance of Content to Computation." *Philosophy and Phenomenological Research* 88: pp. 173-208.
---. 2014b. "A Theory of Computational Implementation." *Synthese* 191: pp. 1277-1307.
---. 2015. "Bayesian Perceptual Psychology." In *The Oxford Handbook of the Philosophy of Perception*, ed. M. Matthen. Oxford: Oxford University Press.
--- 2020a. "Perceptual Co-Reference," *The Review of Philosophy and Psychology* 11: pp. 569-589.
---. 2020b. "A Realist Perspective on Bayesian Cognitive Science." In *Inference and*

Consciousness, eds. A. Nes and T. Chan. New York: Routledge.
---. 2021. "Bayesian Modeling of the Mind: From Norm to Neurons." *WIREs Cognitive Science* 12: e1540.
Rullán Buxó, C., and Savin, C. 2021. "A Sampling-Based Circuit for Optimal Decision-Making." *Advances in Neural Information Processing Systems* 34.
Sanborn, A., and Chater, N. 2016. "Bayesian Brains Without Probabilities." *Trends in Cognitive Science* 20: pp. 883-893.
Sato, Y., Toyoizumi, T., and Aihara, K. 2007. "Bayesian Inference Explains Perception of Unity and Ventriloquism Aftereffect: Identification of Common Sources of Audiovisual Stimuli." *Neural Computation* 19: pp. 3335-3355.
Savin, C., and Denève, S. 2014. "Spatio-temporal Representations of Uncertainty in Spiking Neural Networks." *Advances in Neural Information Processing Systems* 27.
Searle, J. 1990. "Is the Brain a Digital Computer?". *Proceedings and Addresses of the American Philosophical Association* 64: pp. 21-37.
Simoncelli, E. 2009. "Optimal Estimation in Sensory Systems." In *The New Cognitive Neurosciences*, 4th edition, ed. M. Gazzaniga. Cambridge: MIT Press.
Sohn, H., and Narain, D. 2021. "Neural Implementations of Bayesian Inference." *Current Opinion in Neurobiology* 70: pp. 121-129.
Sokoloski, S. 2017. "Implementing a Bayes Filter in a Neural Circuit: The Case of Unknown Stimulus Dynamics." *Neural Computation* 29: pp. 2450-2490.
Spratling, M. 2016. "A Neural Implementation of Bayesian Inference Based on Predictive Coding." *Connection Science* 28: pp. 346-383.
Sotiropoulos, G., Seitz, A., and Seriès, P. 2011. "Changing Expectations about Speed Alters Perceived Motion Direction." *Current Biology* 21: R883-R884.
---. 2017. "A Review of Predictive Coding Algorithms." *Brain and Cognition* 112: pp. 92-97.
Stocker, A., and Simoncelli, E. 2006. "Noise Characteristics and Prior Expectations in Human Visual Speed Perception." *Nature Neuroscience* 4: pp. 578-585.
van Rooij, I., Blokpoel, M., Kwisthout, J., and Wareham, T. 2019. *Cognition and Intractability*. Cambridge: Cambridge University Press.
Walker, E., Cotton, R. J., Ma, W. J., and Tolias, A. 2020. "A Neural Basis of Probabilistic Computation in Visual Cortex." *Nature Neuroscience* 23: pp. 122-129.
Weiss, Y., Simoncelli, E., and Adelson, E. 2002. "Motion Illusions as Optimal Percepts." *Nature Neuroscience* 5: pp. 598-604.
Woodward, J. 2008. "Mental Causation and Neural Mechanisms." In *Being Reduced*, eds. J. Hohwy and J. Kallestrup. Oxford: Oxford University Press.
---. 2015. "Interventionism and Causal Exclusion." Philosophy and Phenomenological Research 91: pp. 303-347.
Zemel, R., and Dayan, P. 1997. "Combining Probabilistic Population Codes." In *JCAI-97: 15th International Joint Conference on Artificial Intelligence*. San Francisco: Morgan Kaufmann.
Zemel, R., Dayan, P., and Pouget, A. 1998. "Probabilistic Interpretation of Population Codes." *Neural Computation* 10: pp. 403-430.

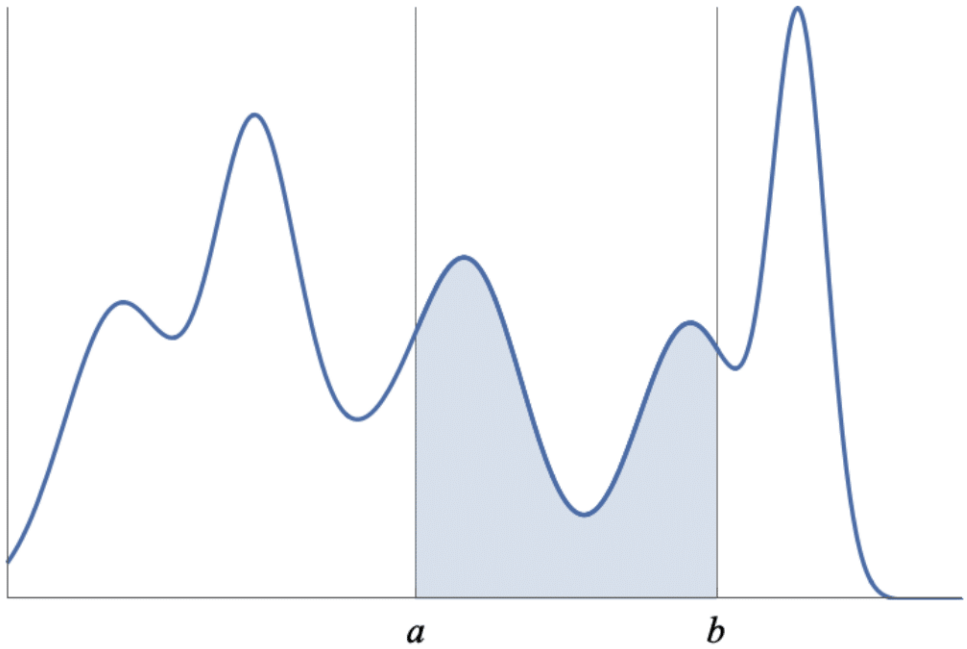

Figure 1. An example of a pdf $p(x)$. The area under the curve between points a and b is the probability assigned to the interval $[a, b]$.

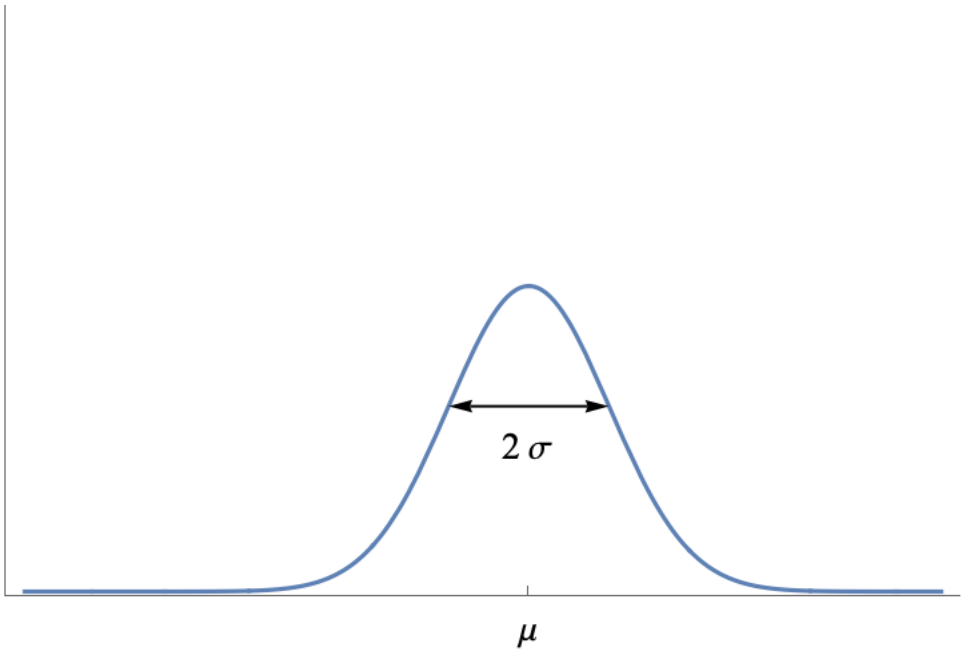

Figure 2. Gaussian pdf with mean μ and variance σ^2.

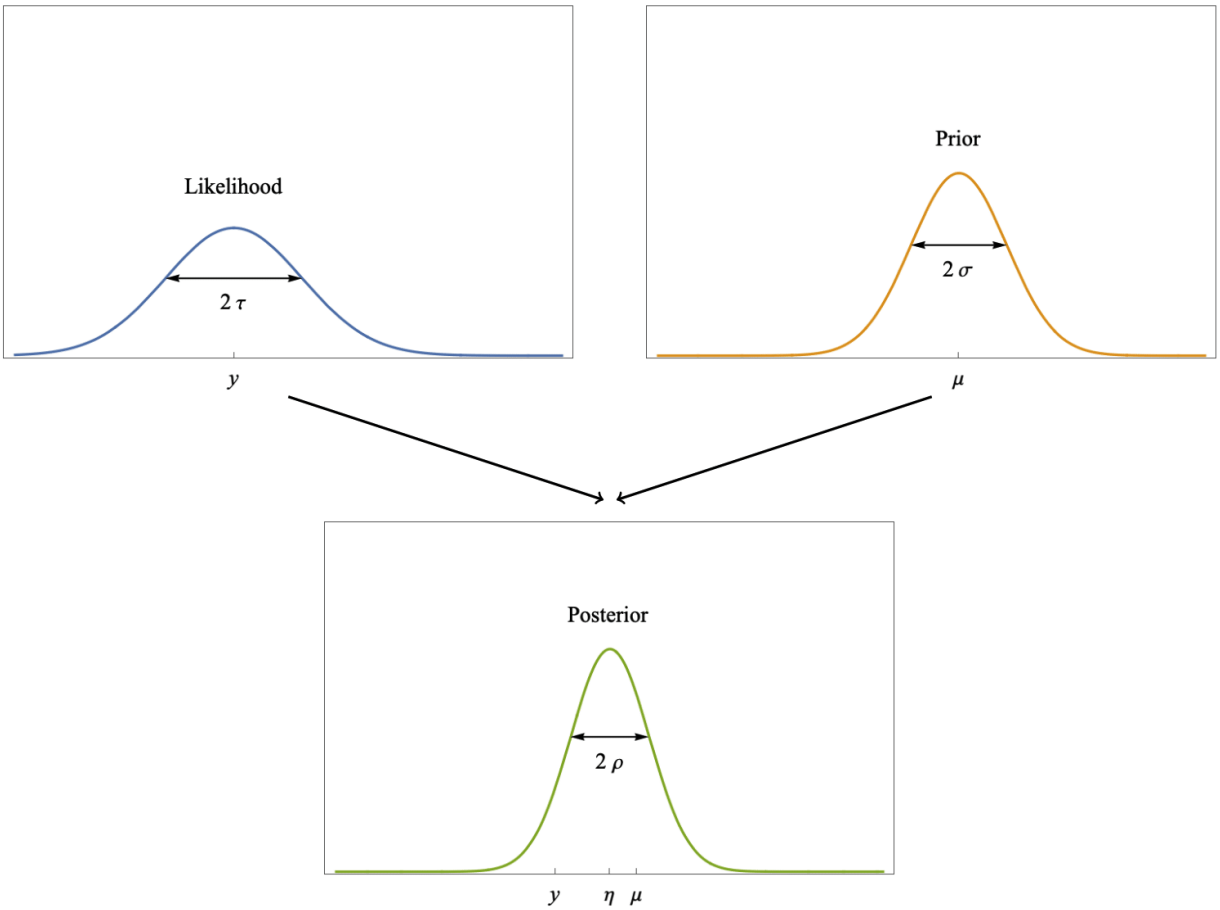

Figure 3. The top left panel is the likelihood function defined by equation (2) for fixed *y*. The top right panel is the pdf defined by equation (1). The bottom panel is the posterior determined by equation (3).

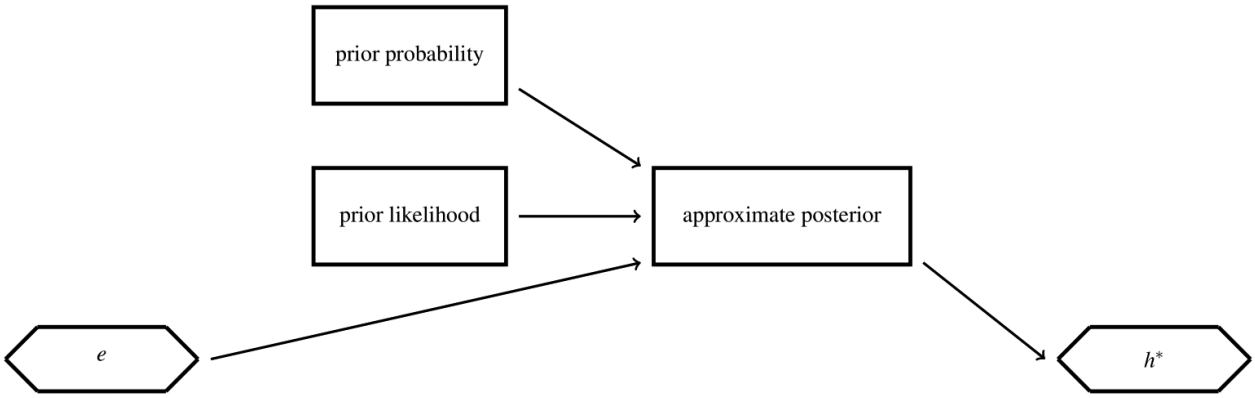

Figure 4. Causal structure of approximate Bayesian inference. Arrows represent the direction of causal influence.

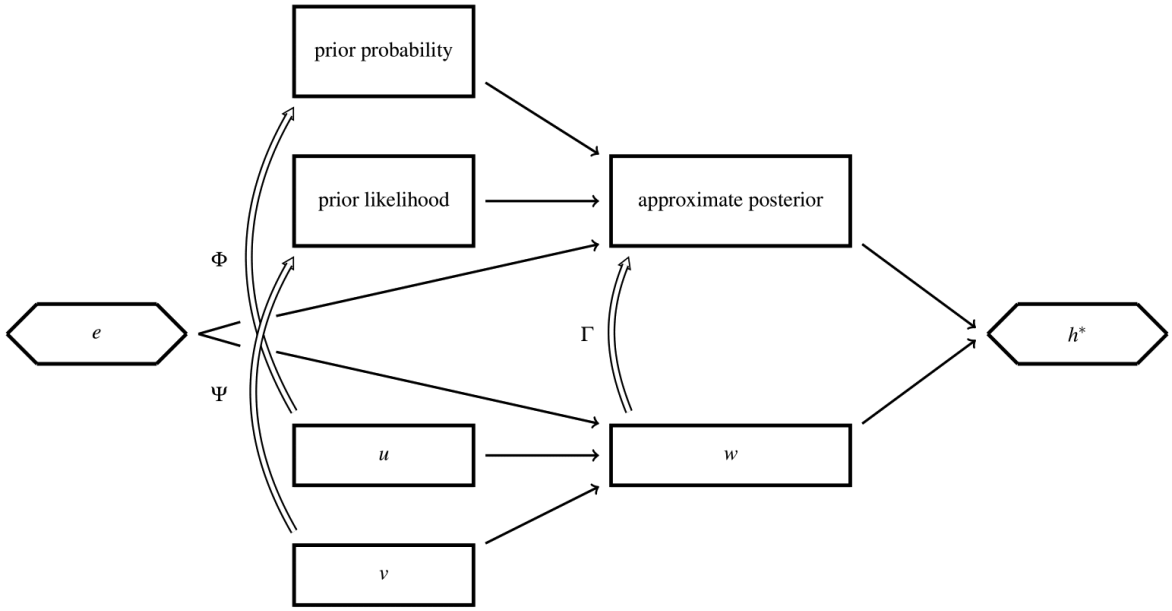

Figure 5. Schematic form for a neural implementation theory. Single arrows represent the direction of causal influence. Double arrows represent decoders, which map neural states to credal states.

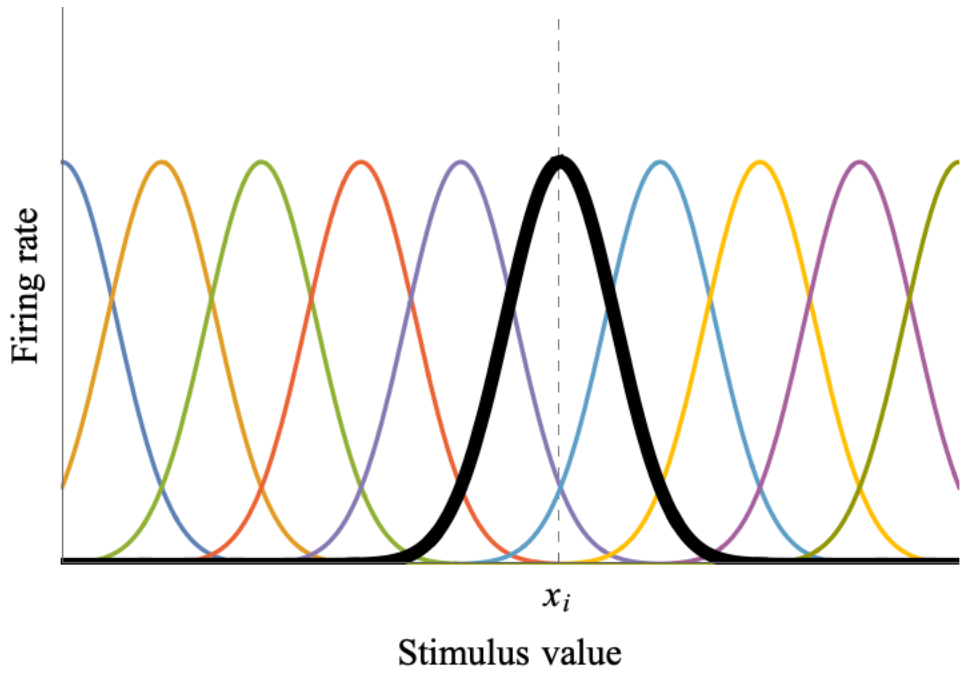

Figure 6. A collection of Gaussian tuning curves. The horizontal axis contains possible values of a one-dimensional continuous stimulus. Each tuning curve depicts the average response of the corresponding neuron to possible stimulus values. $f_i(x)$ is the average firing rate elicited by stimulus x in neuron i. The tuning curve $f_i(x)$ with preferred stimulus value x_i is thickened. Firing rate is typically measured in spikes per second. Shapes and maximum values for $f_i(x)$ vary with the neural population.

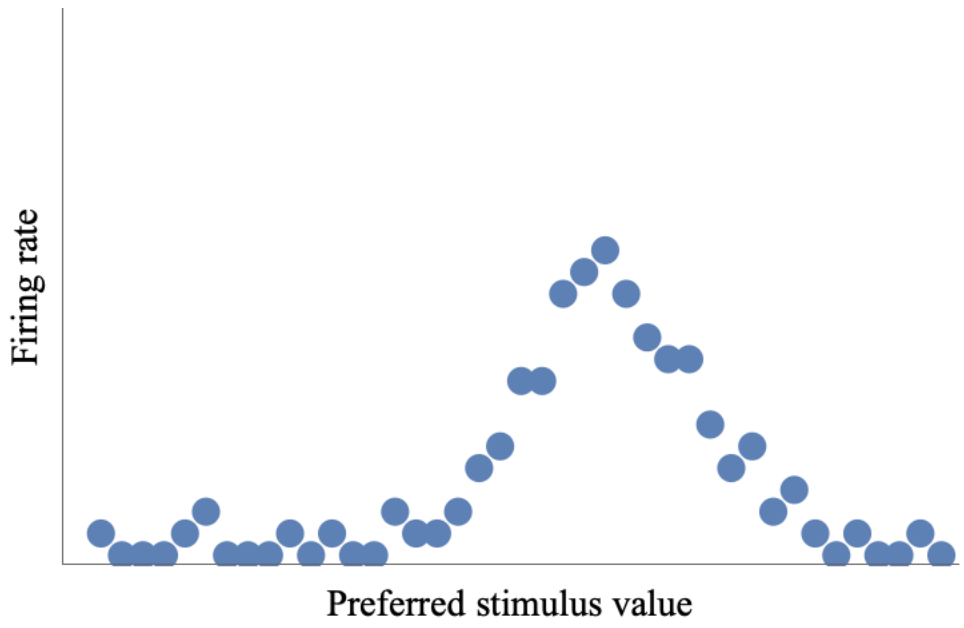

Figure 7. Firing activity in a hypothetical neural population on a given occasion. The horizontal axis groups neurons according to preferred stimulus value. The vertical axis gives the firing rate for each neuron.

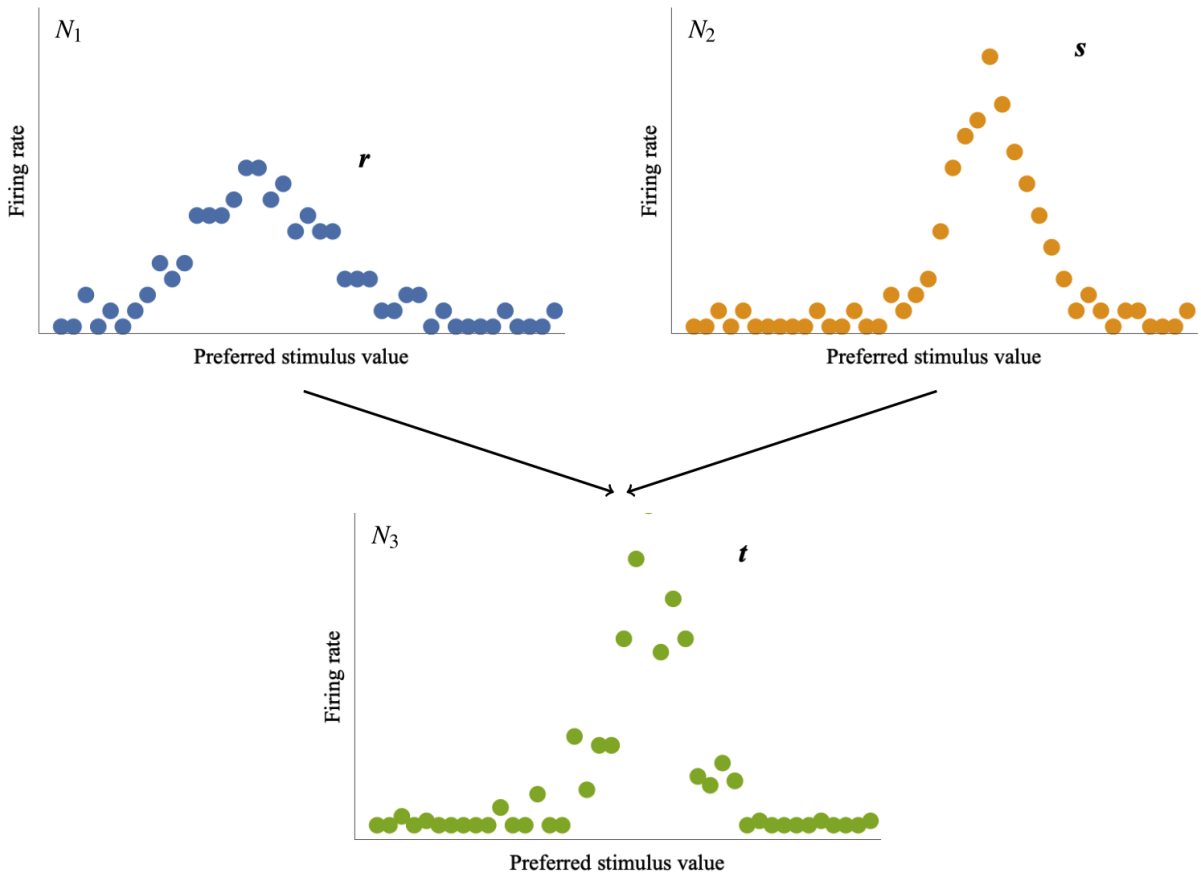

Figure 8. Activity in N_1 (encoding the likelihood function), N_2 (encoding the prior probability), and N_3 (encoding the posterior). In N_3, each neuron's firing rate is obtained by multiplying the firing rates of the corresponding neurons in N_1 and N_2. The vertical axis for N_3 has been rescaled for greater legibility. Note the similarity with Figure 3.

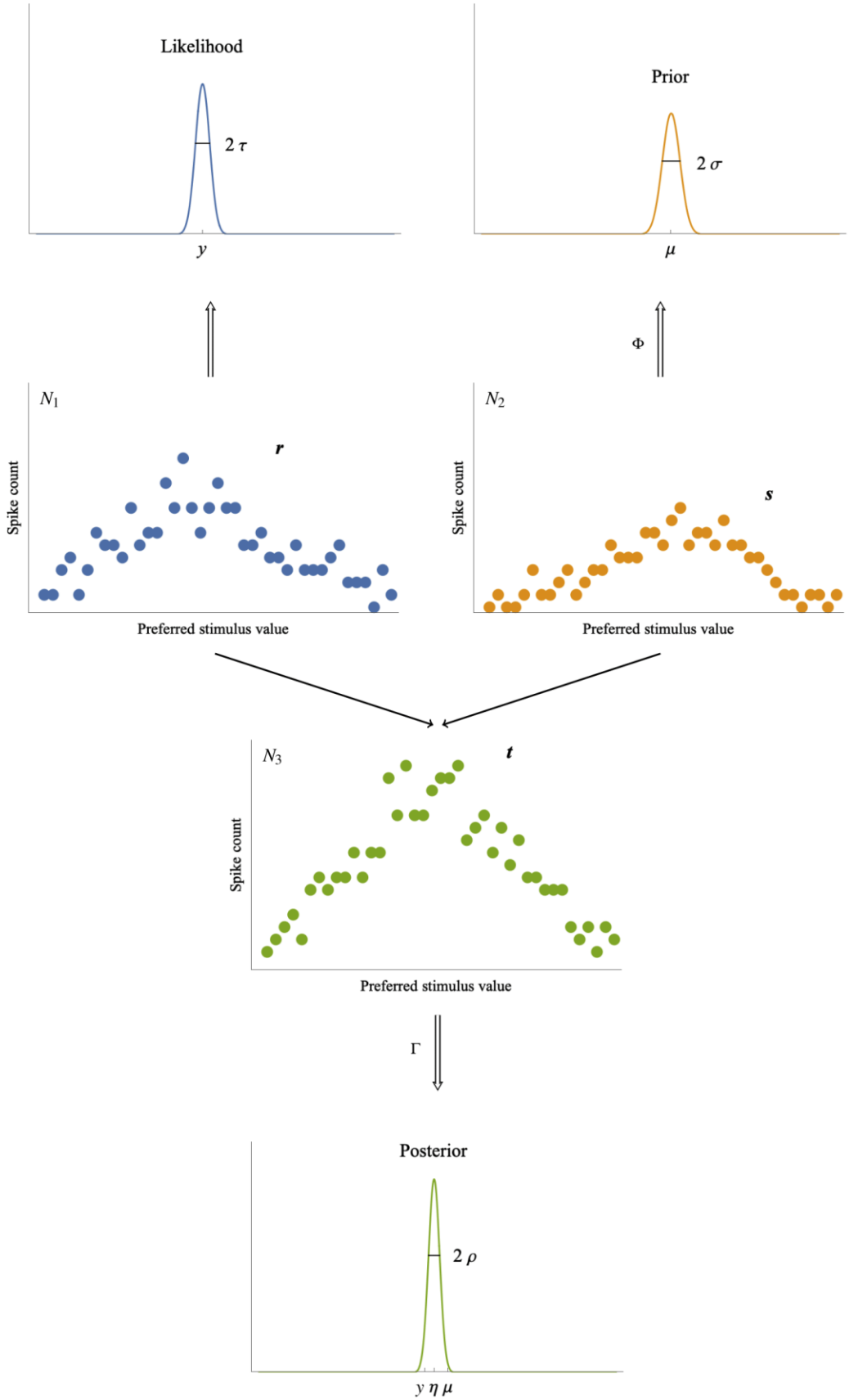
Likelihood
2 τ
y
Prior
2 σ
μ
Φ
N1
r
Spike count
Preferred stimulus value
N2
s
Spike count
Preferred stimulus value
N3
t
Spike count
Preferred stimulus value
Γ
Posterior
2 ρ
y η μ

Figure 9. Activity in hypothetical neural populations N_1, N_2, and N_3. In N_3, each neuron's spike count is obtained by adding the spike counts of the corresponding neurons in N_1 and N_2. The spike count profile $\boldsymbol{r}$ over N_1 encodes an unnormalized Gaussian likelihood with mean y and variance τ^2 given by (15). The spike count profile $\boldsymbol{s}$ over N_2 encodes a Gaussian prior with mean μ and variance σ^2 given by (16). The spike count profile $\boldsymbol{t}$ over N_3 encodes a Gaussian posterior with mean η and variance ρ^2 given by (18). Note that the mapping from $\boldsymbol{r}$ to the unnormalized Gaussian is *not* a decoder Ψ in the sense of (7), because it carries the spike count profile to a likelihood function rather than a prior likelihood.

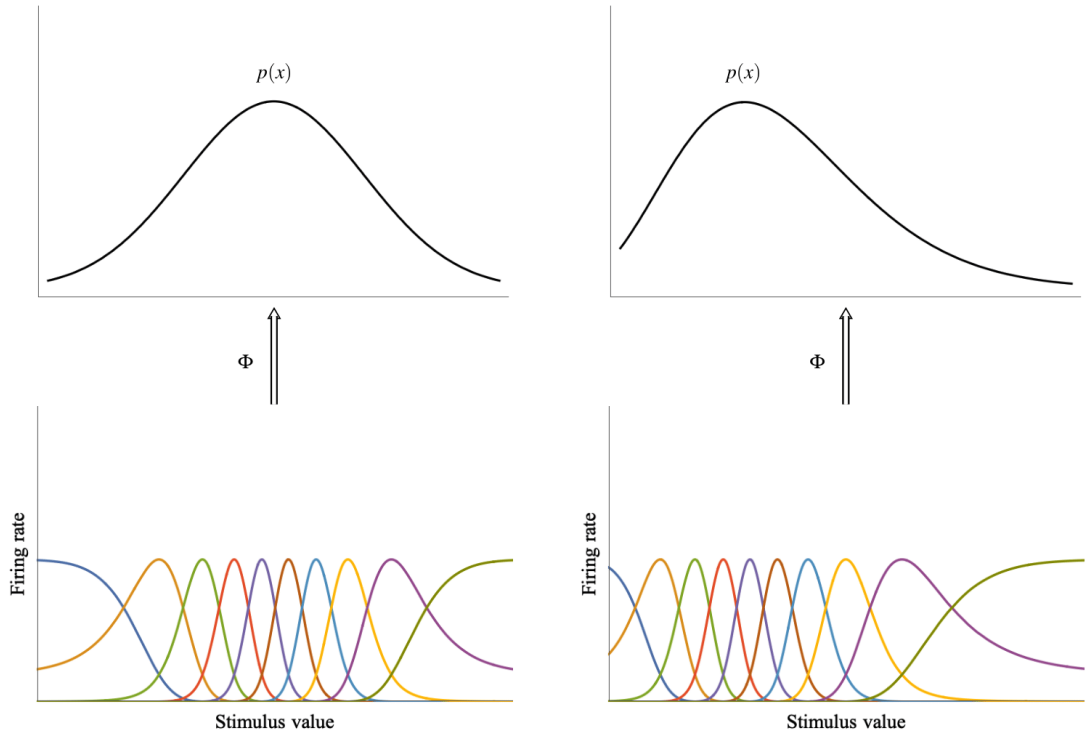

Figure 10. The bottom left panel depicts a collection of tuning curves f_i warped by a density function $d(x)$ via equation (19). The pdf $p(x)$ determined by decoder (20) is depicted in the top left panel. The bottom right panel depicts a collection of tuning curves warped by a different density function. The top right panel depicts the encoded pdf.

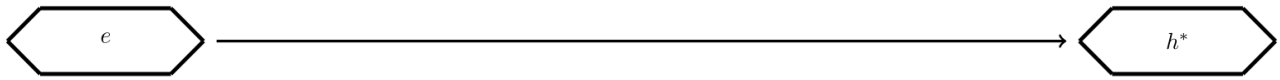

Figure 11. Causal structure of a neural network that does not encode priors or a posterior.

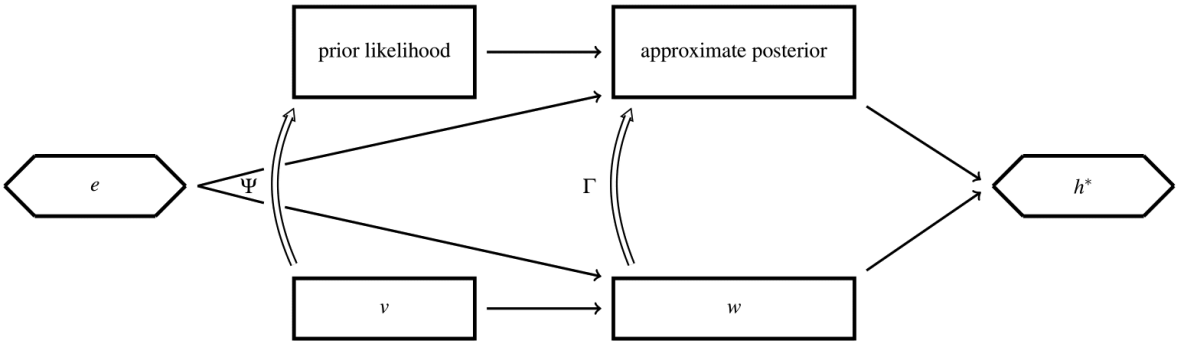

Figure 12. Causal structure of a neural network that encodes a prior likelihood and an approximate posterior but not a prior probability.

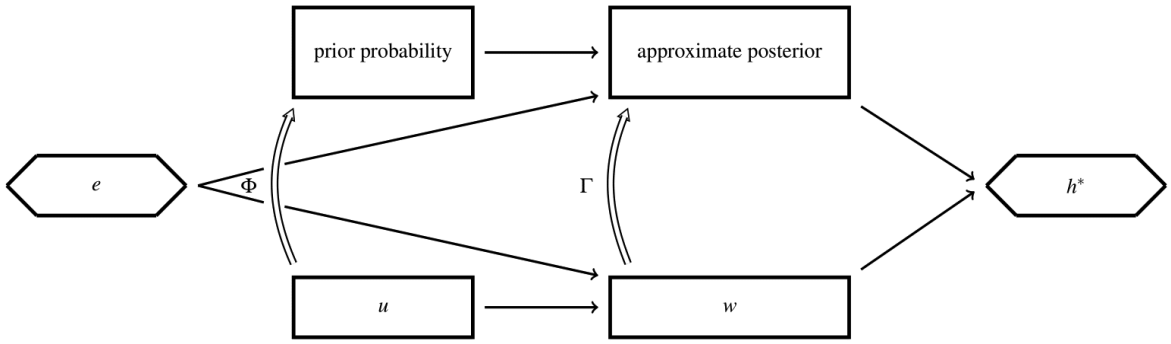

Figure 13. Causal structure of a neural network that encodes a prior probability and an approximate posterior but not a prior likelihood.

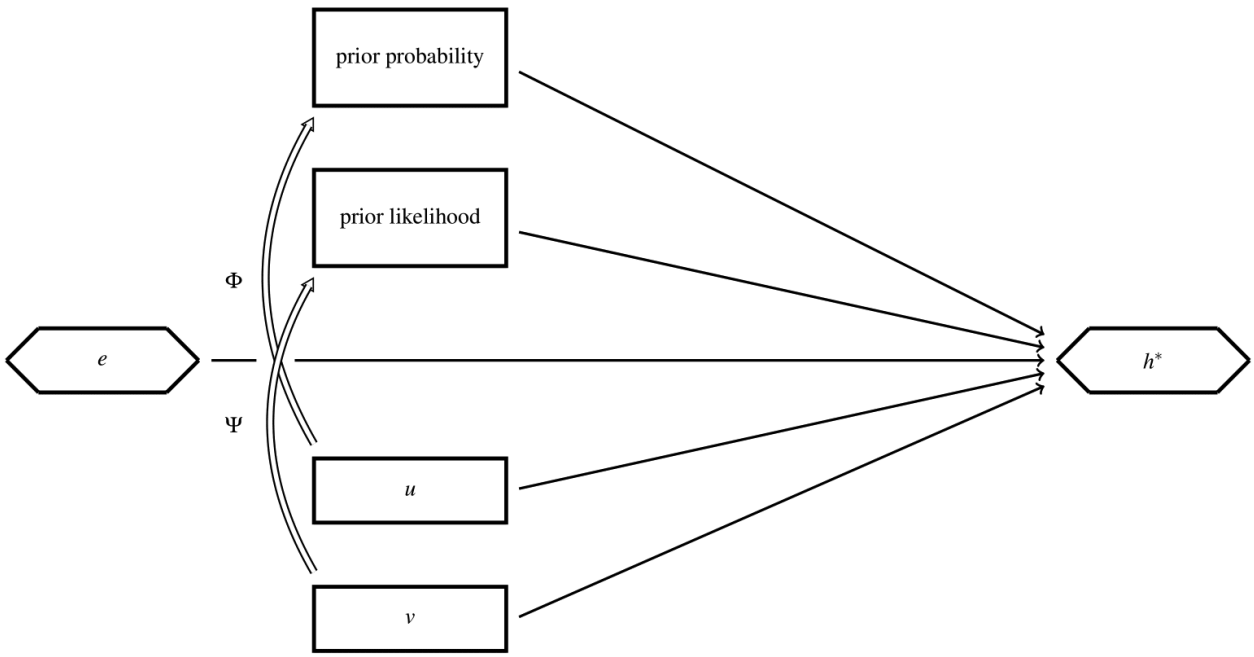

Figure 14. Causal structure of a neural network that encodes a prior probability and a prior likelihood but not an approximate posterior.

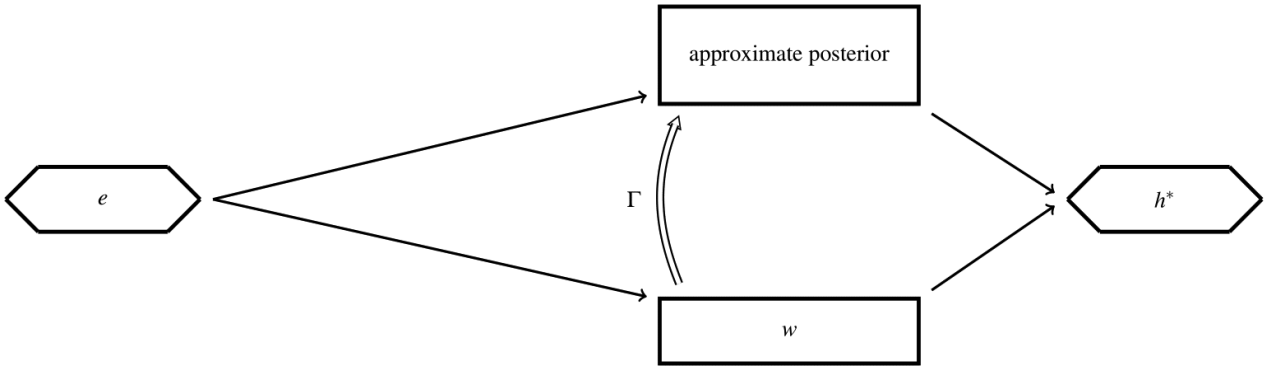

Figure 15. Causal structure of a neural network that encodes an approximate posterior but not a prior probability or a prior likelihood.