

Predictive Coding, Variational Autoencoders, and Biological Connections

Joseph Marino*

josephmarino@deepmind.com

*Computation and Neural Systems, California Institute of Technology,
Pasadena, CA 91125, U.S.A.*

We present a review of predictive coding, from theoretical neuroscience, and variational autoencoders, from machine learning, identifying the common origin and mathematical framework underlying both areas. As each area is prominent within its respective field, more firmly connecting these areas could prove useful in the dialogue between neuroscience and machine learning. After reviewing each area, we discuss two possible correspondences implied by this perspective: cortical pyramidal dendrites as analogous to (nonlinear) deep networks and lateral inhibition as analogous to normalizing flows. These connections may provide new directions for further investigations in each field.

1 Introduction

1.1 Cybernetics. Machine learning and theoretical neuroscience once overlapped under the field of cybernetics (Wiener, 1948; Ashby, 1956). Within this field, perception and control, in both biological and nonbiological systems, were formulated in terms of negative feedback and feedforward processes. Negative feedback attempts to minimize error signals by feeding the errors back into the system, whereas feedforward processing attempts to preemptively reduce error through prediction. Cybernetics formalized these techniques using probabilistic models, which estimate the likelihood of random outcomes, and variational calculus, a technique for estimating functions, particularly probability distributions (Wiener, 1948). This resulted in the first computational models of neuron function and learning (McCulloch & Pitts, 1943; Rosenblatt, 1958; Widrow & Hoff, 1960), a formal definition of information (Wiener, 1942; Shannon, 1948) (with connections to neural systems Barlow, 1961b), and algorithms for negative feedback perception and control (MacKay, 1956; Kalman, 1960). Yet with advances in these directions (see Prieto et al., 2016) the cohesion of cybernetics diminished, with the new ideas taking root in, for

*The author is now at DeepMind, London, U.K.

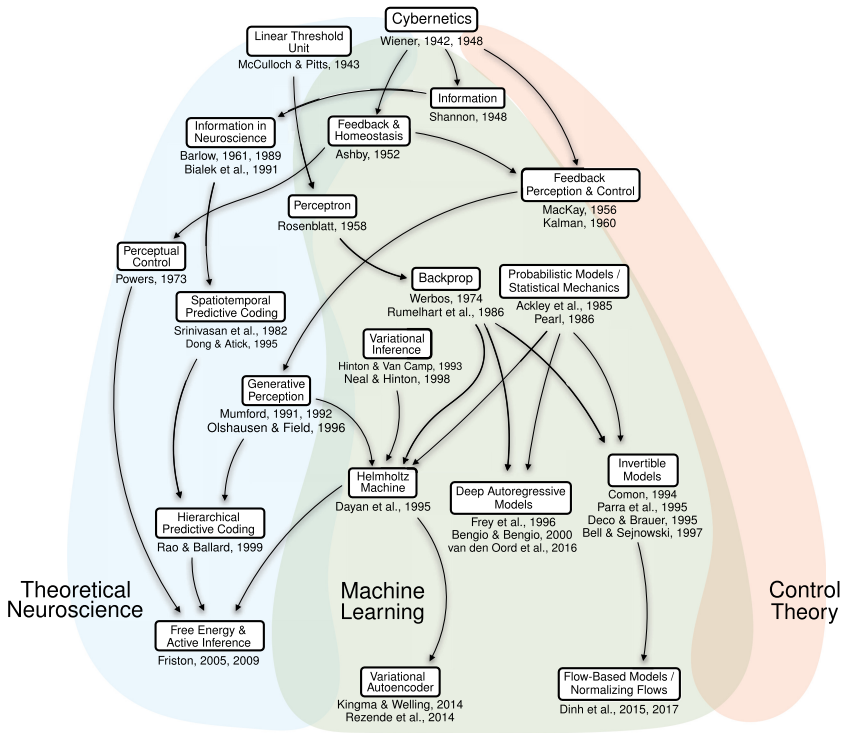


Figure 1: Concept overview. Cybernetics influenced the areas that became theoretical neuroscience and machine learning, resulting in shared mathematical concepts. This review explores the connections between predictive coding, from theoretical neuroscience, and variational autoencoders, from machine learning.

example, theoretical neuroscience, machine learning, and control theory. The transfer of ideas is shown in Figure 1.

1.2 Neuroscience and Machine Learning: Convergence and Divergence. A renewed dialogue between neuroscience and machine learning formed in the 1980s and 1990s. Neuroscientists, bolstered by new physiological and functional analyses, began making traction in studying neural systems in probabilistic and information-theoretic terms (Laughlin, 1981; Srinivasan, Laughlin, & Dubs, 1982; Barlow, 1989; Bialek, Rieke, Van Steveninck, & Warland, 1991). In machine learning, improvements in probabilistic modeling (Pearl, 1986) and artificial neural networks (Rumelhart, Hinton, & Williams, 1986) combined with ideas from statistical mechanics (Hopfield, 1982; Ackley, Hinton, & Sejnowski, 1985) to yield new classes of models and training techniques. This convergence of ideas,

primarily centered around perception, resulted in new theories of neural processing and improvements in their mathematical underpinnings.

In particular, the notion of predictive coding emerged within neuroscience (Srinivasan et al., 1982; Rao & Ballard, 1999). In its most general form, predictive coding postulates that neural circuits are engaged in estimating probabilistic models of other neural activity and sensory inputs, with feedback and feedforward processes playing a central role. These models were initially formulated in early sensory areas, for example, in the retina (Srinivasan et al., 1982) and thalamus (Dong & Atick, 1995), using feedforward processes to predict future neural activity. Similar notions were extended to higher-level sensory processing in neocortex by David Mumford (1991, 1992). Top-down neural projections (from higher-level to lower-level sensory areas) were hypothesized to convey sensory predictions, whereas bottom-up neural projections were hypothesized to convey prediction errors. Through negative feedback, these errors then updated state estimates. These ideas were formalized by Rao and Ballard (1999), formulating a simplified artificial neural network model of images, reminiscent of a Kalman filter (Kalman, 1960).

Feedback and feedforward processes also featured prominently in machine learning. Indeed, the primary training algorithm for artificial neural networks, backpropagation (Rumelhart et al., 1986), literally feeds (propagates) the output prediction errors back through the network—negative feedback. During this period, the technique of variational inference was rediscovered within machine learning (Hinton & Van Camp, 1993; Neal & Hinton, 1998), recasting probabilistic inference using variational calculus. This technique proved essential in formulating the Helmholtz machine (Dayan et al., 1995; Dayan & Hinton, 1996), a hierarchical unsupervised probabilistic model parameterized by artificial neural networks. Similar advances were made in autoregressive probabilistic models (Frey, Hinton, & Dayan, 1996; Bengio & Bengio, 2000), using artificial neural networks to form sequential feedforward predictions, as well as new classes of invertible probabilistic models (Comon, 1994; Parra, Deco, & Miesbach, 1995; Deco & Brauer, 1995; Bell & Sejnowski, 1997).

These new ideas regarding variational inference and probabilistic models, particularly the Helmholtz machine (Dayan, Hinton, Neal, & Zemel, 1995), influenced predictive coding. Specifically, Karl Friston utilized variational inference to formulate hierarchical dynamical models of neocortex (Friston, 2005, 2008a). In line with Mumford (1992), these models contain multiple levels, with each level attempting to predict its future activity (feedforward) as well as lower-level activity, closer to the input data. Prediction errors across levels facilitate updating higher-level estimates (negative feedback). Such models have incorporated many biological aspects, including local learning rules (Friston, 2005) and attention (Spratling, 2008; Feldman & Friston, 2010; Kanai, Komura, Shipp, & Friston, 2015), and have been compared with neural circuits (Bastos et al., 2012; Keller & Mrsic-Flogel,

2018; Walsh, McGovern, Clark, and O’Connell, 2020). While predictive coding and other Bayesian brain theories are increasingly popular (Doya, Ishii, Pouget, & Rao, 2007; Friston, 2009; Clark, 2013), validating these models is hampered by the difficulty of distinguishing between specific design choices and general theoretical claims (Gershman, 2019). Further, a large gap remains between the simplified implementations of these models and the complexity of neural systems.

Progress in machine learning picked up in the early 2010s, with advances in parallel computing as well as standardized data sets (Deng et al., 2009). In this era of deep learning (LeCun, Bengio, & Hinton, 2015; Schmidhuber, 2015), that is, artificial neural networks with multiple layers, a flourishing of ideas emerged around probabilistic modeling. Building off previous work, more expressive classes of deep hierarchical (Gregor, Danihelka, Mnih, Blundell, & Wierstra, 2014; Mnih & Gregor, 2014; Kingma & Welling, 2014; Rezende, Mohamed, & Wierstra, 2014), autoregressive (Uria, Murray, & Larochelle, 2014; van den Oord, Kalchbrenner, & Kavukcuoglu, 2016), and invertible (Dinh, Krueger, & Bengio, 2015; Dinh, Sohl-Dickstein, & Bengio, 2017) probabilistic models were developed. Of particular importance is a model class known as variational autoencoders (VAEs; Kingma & Welling, 2014; Rezende et al., 2014), a relative of the Helmholtz machine, which closely resembles hierarchical predictive coding. Unfortunately, despite this similarity, the machine learning community remains largely oblivious to the progress in predictive coding and vice versa.

1.3 Connecting Predictive Coding and VAEs. This review aims to bridge the divide between predictive coding and VAEs. While this work provides unique contributions, it is inspired by previous work at this intersection. In particular, van den Broeke (2016) outlines hierarchical probabilistic models in predictive coding and machine learning. Likewise, Lotter, Kreiman, and Cox (2017, 2018) implement predictive coding techniques in deep probabilistic models, comparing these models with neural phenomena.

After reviewing background mathematical concepts in section 2, we discuss the basic formulations of predictive coding in section 3 and variational autoencoders in section 4, and we identify commonalities in their model formulations and inference techniques in section 5. Based on these connections, in section 6, we discuss two possible correspondences between machine learning and neuroscience seemingly suggested by this perspective:

- **Dendrites of pyramidal neurons and deep artificial networks**, affirming a more nuanced perspective over the analogy of biological and artificial neurons
- **Lateral inhibition and normalizing flows**, providing a more general framework for normalization.

Like the work of van den Broeke (2016) and Lotter et al. (2017, 2018), we hope that these connections will inspire future research in exploring this promising direction.

2 Background

2.1 Maximum Log Likelihood. Consider a random variable, $\mathbf{x} \in \mathbb{R}^M$, with a corresponding distribution, $p_{\text{data}}(\mathbf{x})$, defining the probability of observing each possible value. This distribution is the result of an underlying data-generating process, for example, the emission and scattering of photons. While we do not have direct access to p_{data} , we can sample observations, $\mathbf{x} \sim p_{\text{data}}(\mathbf{x})$, yielding an empirical distribution, $\hat{p}_{\text{data}}(\mathbf{x})$. Often we wish to model p_{data} , for example, for prediction or compression. We refer to this model as $p_{\theta}(\mathbf{x})$, with parameters θ . Estimating the model parameters involves maximizing the log likelihood of data samples under the model's distribution:

$$\theta^* \leftarrow \arg \max_{\theta} \mathbb{E}_{\mathbf{x} \sim p_{\text{data}}(\mathbf{x})} [\log p_{\theta}(\mathbf{x})]. \quad (2.1)$$

This is the maximum log-likelihood objective, which is found throughout machine learning and probabilistic modeling (Murphy, 2012). In practice, we do not have access to $p_{\text{data}}(\mathbf{x})$ and instead approximate the objective using data samples, that is, using $\hat{p}_{\text{data}}(\mathbf{x})$.

2.2 Probabilistic Models.

2.2.1 Dependency Structure. A probabilistic model includes the dependency structure (see section 2.2.1) and the parameterization of these dependencies (see section 2.2.2). The dependency structure is the set of conditional dependencies between variables (see Figure 2). One common form is given by autoregressive models (Frey et al., 1996; Bengio & Bengio, 2000), which use the chain rule of probability:

$$p_{\theta}(\mathbf{x}) = \prod_{j=1}^M p_{\theta}(x_j | x_{<j}). \quad (2.2)$$

By inducing an ordering over the M dimensions of \mathbf{x} , we can factor the joint distribution, $p_{\theta}(\mathbf{x})$, into a product of M conditional distributions, each conditioned on the previous dimensions, $x_{<j}$. A natural use case arises in modeling sequential data, where time provides an ordering over a sequence of T variables, $\mathbf{x}_{1:T}$:

$$p_{\theta}(\mathbf{x}_{1:T}) = \prod_{t=1}^T p_{\theta}(\mathbf{x}_t | \mathbf{x}_{<t}). \quad (2.3)$$

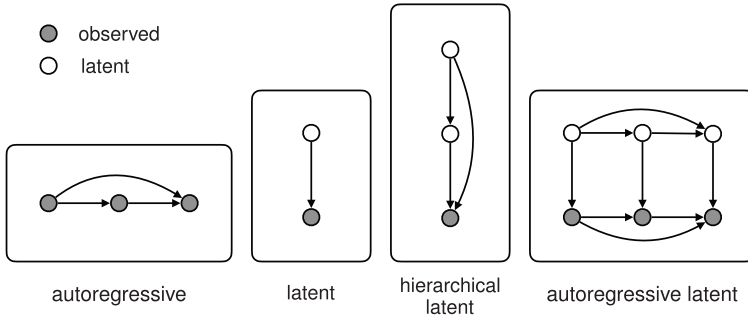


Figure 2: Dependency structures. Each diagram shows a directed graphical model. Nodes represent random variables, and arrows represent dependencies. The main forms of dependency structure are autoregressive (see equation 2.2) and latent variable models (see equation 2.4). These structures can be combined in various ways (see equations 2.7 and 2.8).

Autoregressive models are “fully visible” models (Frey et al., 1996), as dependencies are only modeled between observed variables. However, we can also introduce latent variables, \mathbf{z} . Formally, a latent variable model is defined by the joint distribution

$$p_{\theta}(\mathbf{x}, \mathbf{z}) = p_{\theta}(\mathbf{x}|\mathbf{z})p_{\theta}(\mathbf{z}), \quad (2.4)$$

where $p_{\theta}(\mathbf{x}|\mathbf{z})$ is the conditional likelihood and $p_{\theta}(\mathbf{z})$ is the prior. Introducing latent variables is one of, if not, *the* primary technique for increasing the flexibility of a model, as evaluating the likelihood now requires marginalizing over the latent variables:

$$p_{\theta}(\mathbf{x}) = \mathbb{E}_{\mathbf{z} \sim p_{\theta}(\mathbf{z})} [p_{\theta}(\mathbf{x}|\mathbf{z})]. \quad (2.5)$$

Thus, $p_{\theta}(\mathbf{x})$ is a mixture distribution, with each component, $p_{\theta}(\mathbf{x}|\mathbf{z})$, weighted according to $p_{\theta}(\mathbf{z})$. Even when $p_{\theta}(\mathbf{x}|\mathbf{z})$ takes a simple distribution form, such as gaussian, $p_{\theta}(\mathbf{x})$ can take on flexible forms. In this way, \mathbf{z} can capture complex dependencies in \mathbf{x} .

However, increased flexibility comes with increased computational cost. In general, marginalizing over \mathbf{z} is not tractable. This requires us to adopt approximations, discussed in section 2.3, or restrict the model to ensure tractable evaluation of $p_{\theta}(\mathbf{x})$ (see equation 2.5). The latter approach is taken by flow-based latent variable models (Tabak & Turner, 2013; Rippel & Adams, 2013; Dinh et al., 2015), defining the dependency between \mathbf{x} and \mathbf{z} via an invertible transform, $\mathbf{x} = f_{\theta}(\mathbf{z})$ and $\mathbf{z} = f_{\theta}^{-1}(\mathbf{x})$. With a prior or base distribution, $p_{\theta}(\mathbf{z})$, we can express $p_{\theta}(\mathbf{x})$ using the change of variables formula,

$$p_{\theta}(\mathbf{x}) = p_{\theta}(\mathbf{z}) \left| \det \left(\frac{\partial \mathbf{x}}{\partial \mathbf{z}} \right) \right|^{-1}, \quad (2.6)$$

where $\frac{\partial \mathbf{x}}{\partial \mathbf{z}}$ is the Jacobian of the transform and $\det(\cdot)$ denotes matrix determinant. The term $\left| \det \left(\frac{\partial \mathbf{x}}{\partial \mathbf{z}} \right) \right|^{-1}$ is the local scaling of space when moving from \mathbf{z} to \mathbf{x} , conserving probability mass. Flow-based models, also referred to as *normalizing flows* (Rezende & Mohamed, 2015), are the basis of independent components analysis (ICA) (Comon, 1994; Bell & Sejnowski, 1997; Hyvärinen & Oja, 2000) and nonlinear generalizations (Chen & Gopinath, 2001; Laparra, Camps-Valls, & Malo, 2011). These models provide a general-purpose mechanism for adding and removing dependencies between variables (i.e., normalization).¹ Yet while flow-based models avoid marginalization, their requirement of invertibility can be overly restrictive (Cornish, Caterini, Deligiannidis, & Doucet, 2020).

We have presented autoregression and latent variables separately; however, these techniques can be combined. For instance, hierarchical latent variable models (Dayan et al., 1995) incorporate autoregressive dependencies between latent variables. Considering L levels of latent variables, $\mathbf{z}^{1:L} = [\mathbf{z}^1, \dots, \mathbf{z}^L]$, we can express the joint distribution as

$$p_{\theta}(\mathbf{x}, \mathbf{z}^{1:L}) = p_{\theta}(\mathbf{x} | \mathbf{z}^{1:L}) \prod_{\ell=1}^L p_{\theta}(\mathbf{z}^{\ell} | \mathbf{z}^{\ell+1:L}). \quad (2.7)$$

We can also incorporate latent variables within sequential (autoregressive) models, giving rise to sequential latent variable models. Considering a single level of latent variables in a corresponding sequence, $\mathbf{z}_{1:T}$, we have the following joint distribution:

$$p_{\theta}(\mathbf{x}_{1:T}, \mathbf{z}_{1:T}) = \prod_{t=1}^T p_{\theta}(\mathbf{x}_t | \mathbf{x}_{<t}, \mathbf{z}_{\leq t}) p_{\theta}(\mathbf{z}_t | \mathbf{x}_{<t}, \mathbf{z}_{<t}). \quad (2.8)$$

This encompasses special cases, such as hidden Markov models or linear state-space models (Murphy, 2012). There are a variety of other ways to combine autoregression and latent variables (Gulrajani et al., 2017; Razavi, van den Oord, & Vinyals, 2019). In some cases, autoregressive and flow-based latent variable models are even equivalent (Kingma et al., 2016).

¹Formally, we refer to normalization as one or more steps of a process transforming the data density into a standard gaussian (i.e., Normal), which is equivalent to ICA (Hyvärinen & Oja, 2000). This is a form of redundancy reduction, removing statistical dependencies between data dimensions.

2.2.2 *Parameterizing the Model.* The distributions defining probabilistic dependencies are functions. In this section, we discuss forms that these functions may take. The canonical example is the gaussian (or Normal) distribution, $\mathcal{N}(x; \mu, \sigma^2)$, which is defined by a mean, μ , and variance, σ^2 . This can be extended to the multivariate setting, where $\mathbf{x} \in \mathbb{R}^M$ is modeled with a mean vector, $\boldsymbol{\mu}$, and covariance matrix, $\boldsymbol{\Sigma}$, with the probability density written as

$$\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{M/2} \det(\boldsymbol{\Sigma})^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x} - \boldsymbol{\mu})\right). \quad (2.9)$$

To simplify calculations, we may consider diagonal covariance matrices, $\boldsymbol{\Sigma} = \text{diag}(\boldsymbol{\sigma}^2)$. In particular, the special case where $\boldsymbol{\Sigma} = \mathbf{I}_M$, the $M \times M$ identity matrix, the log-density becomes the mean squared error:

$$\log \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \mathbf{I}) = -\frac{1}{2} \|\mathbf{x} - \boldsymbol{\mu}\|_2^2 + \text{const}. \quad (2.10)$$

Conditional dependencies are mediated by the distribution parameters, which are functions of the conditioning variables. For example, we can express an autoregressive gaussian distribution (see equation 2.2) through $p_\theta(x_j | x_{<j}) = \mathcal{N}(x_j; \mu_\theta(x_{<j}), \sigma_\theta^2(x_{<j}))$, where μ_θ and σ_θ^2 are functions taking $x_{<j}$ as input. A similar form applies to autoregressive models on sequences of vector inputs (see equation 2.3), with $p_\theta(\mathbf{x}_t | \mathbf{x}_{<t}) = \mathcal{N}(\mathbf{x}_t; \boldsymbol{\mu}_\theta(\mathbf{x}_{<t}), \boldsymbol{\Sigma}_\theta(\mathbf{x}_{<t}))$. Likewise, in a latent variable model (see equation 2.4), we can express a gaussian conditional likelihood as $p_\theta(\mathbf{x} | \mathbf{z}) = \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_\theta(\mathbf{z}), \boldsymbol{\Sigma}_\theta(\mathbf{z}))$. In the above examples, we have used a subscript θ for all functions; however, these may be separate functions in practice.

The functions supplying each of the distribution parameters can range in complexity, from constant to highly nonlinear. Classical modeling techniques often employ linear functions. For instance, in a latent variable model, we could parameterize the mean as

$$\boldsymbol{\mu}_\theta(\mathbf{z}) = \mathbf{W}\mathbf{z} + \mathbf{b}, \quad (2.11)$$

where \mathbf{W} is a matrix of weights and \mathbf{b} is a bias vector. Models of this form underlie factor analysis, probabilistic principal components analysis (Tipping & Bishop, 1999), independent components analysis (Bell & Sejnowski, 1997; Hyvärinen & Oja, 2000), and sparse coding (Olshausen & Field, 1996). While linear models are computationally efficient, they are often too limited to accurately model complex data distributions, such as those found in natural images or audio.

Deep learning (Goodfellow, Bengio, & Courville, 2016) provides probabilistic models with expressive nonlinear functions, improving their capacity. In these models, the distribution parameters are parameterized

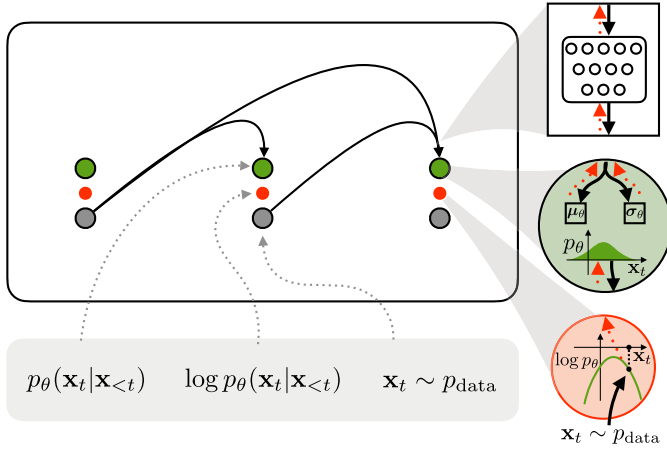


Figure 3: Autoregressive computation graph. The graph contains the (gaussian) conditional likelihoods (green), data (gray), and terms in the objective (red dots). Gradients (red dotted lines) backpropagate through the networks parameterizing the distributions.

with deep networks, which are then trained by backpropagating (Rumelhart et al., 1986) the gradient of the log-likelihood, $\nabla_{\theta} \mathbb{E}_{\mathbf{x} \sim \hat{p}_{\text{data}}} [\log p_{\theta}(\mathbf{x})]$, through the network. Deep probabilistic models have enabled recent advances in speech (Graves, 2013; van den Oord et al., 2016), natural language (Sutskever, Vinyals, & Le, 2014; Radford et al., 2019), images (Razavi et al., 2019), video (Kumar et al., 2020), reinforcement learning (Chua, Calandra, McAllister, & Levine, 2018; Ha & Schmidhuber, 2018) and other areas.

In Figure 3, we visualize a computation graph for a deep autoregressive model, breaking the variables into their distributions and terms in the objective. Green circles denote the (gaussian) conditional likelihoods at each step, which are parameterized by deep networks. The log likelihood, $\log p_{\theta}(\mathbf{x}_t | \mathbf{x}_{<t})$, evaluated at the data observation, $\mathbf{x}_t \sim p_{\text{data}}(\mathbf{x}_t | \mathbf{x}_{<t})$ (gray), provides the objective (red dot). The gradient of this objective with regard to the network parameters is calculated through backpropagation (red dotted line).

Autoregressive models have proven useful in many domains. However, there are reasons to prefer latent variable models in some contexts. First, autoregressive sampling is inherently sequential, becoming costly in high-dimensional domains. Second, latent variables provide a representation for downstream tasks, compression, and overall data analysis. Finally, latent variables increase flexibility, which is useful for modeling complex distributions with relatively simple (e.g., gaussian) conditional distributions. While flow-based latent variable models offer one option, their invertibility

requirement limits the types of functions that can be used. For these reasons, we require methods for handling the latent marginalization in equation 2.5. Variational inference is one such method.

2.3 Variational Inference. Training latent variable models through maximum likelihood requires evaluating $\log p_\theta(\mathbf{x})$. However, evaluating $p_\theta(\mathbf{x}) = \int p_\theta(\mathbf{x}, \mathbf{z}) d\mathbf{z}$ is generally intractable. Thus, we require some technique for approximating $\log p_\theta(\mathbf{x})$. Variational inference (Hinton & Van Camp, 1993; Jordan, Ghahramani, Jaakkola, & Saul, 1998) approaches this problem by introducing an approximate posterior distribution, $q(\mathbf{z}|\mathbf{x})$, which provides a lower bound, $\mathcal{L}(\mathbf{x}; q, \theta) \leq \log p_\theta(\mathbf{x})$. This lower bound is referred to as the evidence (or variational) lower bound (ELBO), as well as the (negative) free energy. By tightening and maximizing the ELBO with regard to the model parameters, θ , we can approximate maximum likelihood training.

Variational inference converts probabilistic inference into an optimization problem. Given a family of distributions, \mathcal{Q} (e.g., gaussian), variational inference attempts to find the distribution, $q \in \mathcal{Q}$, that minimizes $D_{\text{KL}}(q(\mathbf{z}|\mathbf{x})||p_\theta(\mathbf{z}|\mathbf{x}))$, where $p_\theta(\mathbf{z}|\mathbf{x})$ is the posterior distribution, $p_\theta(\mathbf{z}|\mathbf{x}) = \frac{p_\theta(\mathbf{x}, \mathbf{z})}{p_\theta(\mathbf{x})}$. Because $p_\theta(\mathbf{z}|\mathbf{x})$ includes the intractable $p_\theta(\mathbf{x})$, we cannot minimize this KL divergence directly. Instead, we can rewrite this as

$$D_{\text{KL}}(q(\mathbf{z}|\mathbf{x})||p_\theta(\mathbf{z}|\mathbf{x})) = \log p_\theta(\mathbf{x}) - \mathcal{L}(\mathbf{x}; q, \theta), \quad (2.12)$$

where, in equation 2.12 (see appendix A), we have defined $\mathcal{L}(\mathbf{x}; q, \theta)$, as

$$\mathcal{L}(\mathbf{x}; q, \theta) \equiv \mathbb{E}_{\mathbf{z} \sim q(\mathbf{z}|\mathbf{x})} [\log p_\theta(\mathbf{x}, \mathbf{z}) - \log q(\mathbf{z}|\mathbf{x})] \quad (2.13)$$

$$= \mathbb{E}_{\mathbf{z} \sim q(\mathbf{z}|\mathbf{x})} [\log p_\theta(\mathbf{x}|\mathbf{z})] - D_{\text{KL}}(q(\mathbf{z}|\mathbf{x})||p_\theta(\mathbf{z})). \quad (2.14)$$

Rearranging terms in equation 2.12, we have

$$\log p_\theta(\mathbf{x}) = \mathcal{L}(\mathbf{x}; q, \theta) + D_{\text{KL}}(q(\mathbf{z}|\mathbf{x})||p_\theta(\mathbf{z}|\mathbf{x})). \quad (2.15)$$

Because KL divergence is nonnegative, $\mathcal{L}(\mathbf{x}; q, \theta) \leq \log p_\theta(\mathbf{x})$, with equality when $q(\mathbf{z}|\mathbf{x}) = p_\theta(\mathbf{z}|\mathbf{x})$. As the left-hand side of equation 2.15 does not depend on $q(\mathbf{z}|\mathbf{x})$, maximizing $\mathcal{L}(\mathbf{x}; q, \theta)$ with regard to q implicitly minimizes $D_{\text{KL}}(q(\mathbf{z}|\mathbf{x})||p_\theta(\mathbf{z}|\mathbf{x}))$ with regard to q . Thus, maximizing $\mathcal{L}(\mathbf{x}; q, \theta)$ with regard to q tightens the lower bound on $\log p_\theta(\mathbf{x})$. With this tightened lower bound, we can then maximize $\mathcal{L}(\mathbf{x}; q, \theta)$ with regard to θ . This alternating optimization process is the variational expectation maximization (EM) algorithm (Dempster, Laird, & Rubin, 1977; Neal & Hinton, 1998), consisting of approximate inference (E-step) and learning (M-step).

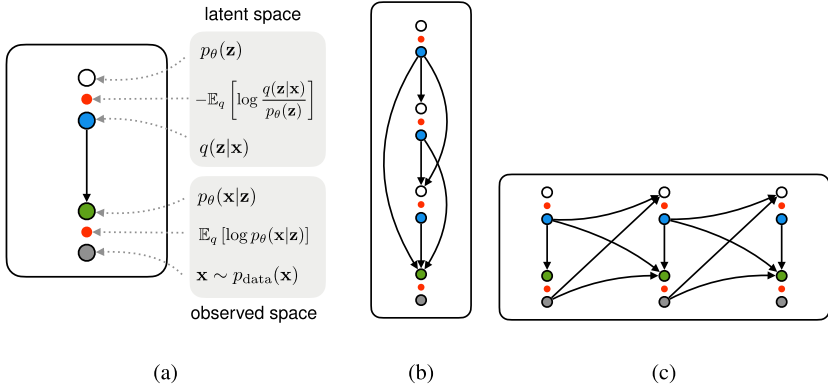


Figure 4: ELBO computation graphs. (a) Basic computation graph for variational inference. Outlined circles denote distributions, smaller red circles denote terms in the ELBO, and arrows denote conditional dependencies. This notation can be used to express (b) hierarchical and (c) sequential models with various dependencies.

We can also represent the ELBO in latent variable models as a computation graph (see Figure 4). Each variable contains a red circle, denoting a term in the ELBO. As compared with the log-likelihood objective, we now have an additional objective term corresponding to the KL divergence for the latent variable. We visualize the variational objective for more complex hierarchical and sequential models in Figures 4b and 4c.

3 Predictive Coding

Predictive coding can be divided into two settings, spatiotemporal and hierarchical, roughly corresponding to the two main forms of probabilistic dependencies. In this section, we review these settings, discussing existing hypothesized correspondences with neural anatomy. We then outline the empirical support for predictive coding, highlighting the need for large-scale, testable models.

3.1 Spatiotemporal Predictive Coding. Spatiotemporal predictive coding (Srinivasan et al., 1982) forms predictions across spatial dimensions and temporal sequences. These predictions produce the resulting “code” as the prediction error. In the temporal setting, we can consider a gaussian autoregressive model defined over observation sequences, $\mathbf{x}_{1:T}$. The conditional probability at time t is

$$p_\theta(\mathbf{x}_t | \mathbf{x}_{<t}) = \mathcal{N}(\mathbf{x}_t; \boldsymbol{\mu}_\theta(\mathbf{x}_{<t}), \text{diag}(\boldsymbol{\sigma}_\theta^2(\mathbf{x}_{<t}))).$$

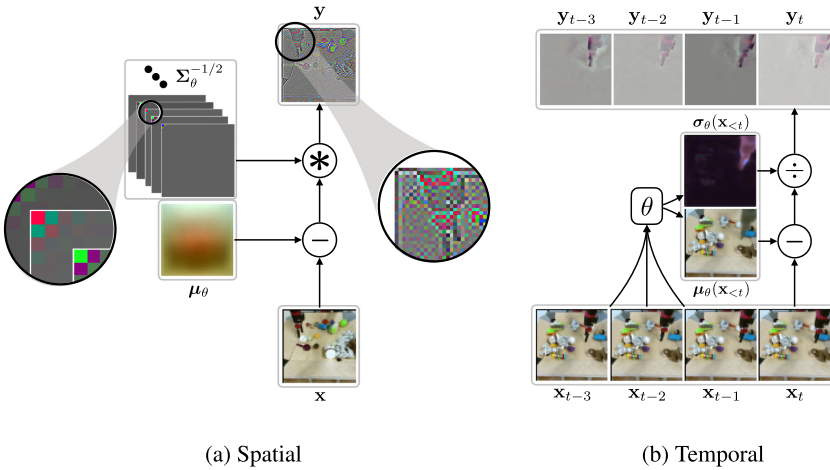


Figure 5: Spatiotemporal predictive coding. (a) Spatial predictive coding removes spatial dependencies, using center-surround filters (left) to extract edges (right). (b) Temporal predictive coding removes temporal dependencies, extracting motion from video. Video frames are from BAIR Robot Pushing (Ebert, Finn, Lee, & Levine, 2017).

Using auxiliary variables, $\mathbf{y}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, we can express $\mathbf{x}_t = \boldsymbol{\mu}_\theta(\mathbf{x}_{<t}) + \boldsymbol{\sigma}_\theta(\mathbf{x}_{<t}) \odot \mathbf{y}_t$, where \odot denotes element-wise multiplication. Conversely, we can express the inverse, normalization or whitening transform as

$$\mathbf{y}_t = \frac{\mathbf{x}_t - \boldsymbol{\mu}_\theta(\mathbf{x}_{<t})}{\boldsymbol{\sigma}_\theta(\mathbf{x}_{<t})}, \quad (3.1)$$

which is a weighted prediction error.² A video example is shown in Figure 5b. The normalization transform removes temporal redundancy in the input, enabling the resulting sequence, $\mathbf{y}_{1:T}$, to be compressed more efficiently (Shannon, 1948; Harrison, 1952; Oliver, 1952). This technique forms the basis of modern video (Wiegand, Sullivan, Bjontegaard, & Luthra, 2003) and audio (Atal & Schroeder, 1979) compression. Note that one special case of this transform is $\boldsymbol{\mu}_\theta(\mathbf{x}_{<t}) = \mathbf{x}_{t-1}$ and $\boldsymbol{\sigma}_\theta(\mathbf{x}_{<t}) = \mathbf{1}$, in which case, $\mathbf{y}_t = \mathbf{x}_t - \mathbf{x}_{t-1} = \Delta \mathbf{x}_t$, that is, temporal changes. For slowly changing sequences, this is a reasonable choice.

Normalization can also be applied within \mathbf{x}_t to remove spatial dependencies. For instance, we can apply another autoregressive transform over spatial dimensions, predicting the i th dimension, $x_{i,t}$, as a function of previous

²Note that other forms of probabilistic models will result in other forms of whitening transforms.

dimensions, $\mathbf{x}_{1:i,t}$ (see equation 2.2). With linear functions, this corresponds to Cholesky whitening (Pourahmadi, 2011; Kingma et al., 2016). However, this imposes an ordering over dimensions. Zero-phase components analysis (ZCA) whitening instead learns symmetric spatial dependencies (Kessy, Lewin, & Strimmer, 2018). Modeling these dependencies with a constant covariance matrix, Σ_θ , and mean, μ_θ , the whitening transform is $\mathbf{y} = \Sigma_\theta^{-1/2}(\mathbf{x} - \mu_\theta)$. With natural images, this results in center-surround filters in the rows of Σ_θ^{-1} , thereby extracting edges (see Figure 5a).

Srinivasan et al. (1982) investigated spatiotemporal predictive coding in the retina, where compression is essential for transmission through the optic nerve. Estimating the (linear) autocorrelation of input sensory signals, they showed that spatiotemporal predictive coding models retinal ganglion cell responses in flies. This scheme allows these neurons to more fully utilize their dynamic range. It is generally accepted that retina, in part, performs stages of spatiotemporal normalization through center-surround receptive fields and on-off responses (Hosoya, Baccus, & Meister, 2005; Graham, Chandler, & Field, 2006; Pitkow & Meister, 2012; Palmer, Marre, Berry, & Bialek, 2015). Dong and Atick (1995) applied similar ideas to the thalamus, proposing an additional stage of temporal normalization. This also relates to the notion of generalized coordinates (Friston, 2008a), that is, modeling temporal derivatives, which can be approximated using finite differences (prediction errors). That is, $\frac{dx}{dt} \approx \Delta \mathbf{x}_t \equiv \mathbf{x}_t - \mathbf{x}_{t-1}$. Thus, spatiotemporal predictive coding may be utilized at multiple stages of sensory processing to remove redundancy (Huang & Rao, 2011).

In neural circuits, normalization often involves inhibitory interneurons (Carandini & Heeger, 2012), performing operations similar to those in equation 3.1. For instance, inhibition occurs in retina between photoreceptors, via horizontal cells, and between bipolar cells, via amacrine cells. This can extract unpredicted motion, e.g., an object moving relative to the background (Ölveczky, Baccus, & Meister, 2003; Baccus, Ölveczky, Manu, & Meister, 2008). A similar scheme is present in the lateral geniculate nucleus (LGN) in thalamus, with interneurons inhibiting relay cells from retina (Sherman & Guillery, 2002). As mentioned above, this is thought to perform temporal normalization (Dong and Atick, 1995; Dan, Atick, & Reid, 1996). Lateral inhibition is also prominent in neocortex, with distinct classes of interneurons shaping the responses of pyramidal neurons (Isaacson & Scanziani, 2011). Part of their computational role appears to be spatiotemporal normalization (Carandini & Heeger, 2012).

3.2 Hierarchical Predictive Coding. Hierarchical predictive coding has been postulated as a model of hierarchical processing in neocortex (Rao & Ballard, 1999; Friston, 2005), the outer sheet-like structure involved in sensory and motor processing (see Figure 6). Neocortex is composed of six layers (I–VI), with neurons arranged into columns, each engaged in related

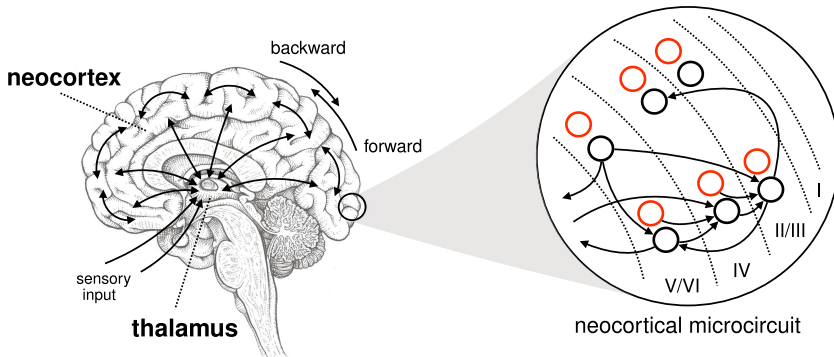


Figure 6: Brain anatomy and cortical circuitry. Sensory inputs enter the thalamus, forming reciprocal connections with the neocortex, which is composed of six layers, with columns across layers and hierarchies of columns. Black and red circles represent excitatory and inhibitory neurons, respectively, with arrows denoting connections. This circuit is repeated with variations throughout neocortex.

computations (Mountcastle, Berman, & Davies, 1955). Columns interact locally via inhibitory interneurons, while also forming longer-range hierarchies via pyramidal neurons. Such hierarchies characterize multiple perceptual (and motor) processing pathways (Van Essen & Maunsell, 1983). Longer-range connections are split into forward (up the hierarchy) and backward (down) directions. Forward connections are driving, evoking neural activity (Girard & Bullier, 1989; Girard, Salin, & Bullier, 1991). Backward connections can be modulatory or driving (Covic & Sherman, 2011; De Pasquale & Sherman, 2011), which can be inverted through inhibition (Meyer et al., 2011). These connections, repeated with variations throughout neocortex, constitute a canonical neocortical microcircuit (Douglas, Martin, & Whitteridge, 1989), suggesting a single algorithm (Hawkins & Blakeslee, 2004), capable of adapting to various inputs (Sharma, Angelucci, & Sur, 2000).

Formulating a theory of neocortex, Mumford (1992) described the thalamus as an “active blackboard,” with the neocortex attempting to reconstruct the activity in the thalamus and lower hierarchical areas. Under this theory, backward projections convey predictions, while forward projections use prediction errors to update estimates. Through a dynamic process, the system settles to an activity pattern, minimizing prediction error. Over time, the parameters are also adjusted to improve predictions. In this way, negative feedback is used, both in inference and learning, to construct a generative model of sensory inputs. Generative state estimation dates back (at least) to Helmholtz (Von Helmholtz, 1867), and error-based updating

is in line with cybernetics (Wiener, 1948; MacKay, 1956), which influenced Kalman filtering (Kalman, 1960), a ubiquitous Bayesian filtering algorithm.

A mathematical formulation of Mumford's model, with ties to Kalman filtering (Rao, 1998), was provided by Rao and Ballard (1999), with the generalization to variational inference provided by Friston (2005). To illustrate this setup, consider a model consisting of a single level of continuous latent variables, \mathbf{z} , modeling continuous data observations, \mathbf{x} . We will use gaussian densities for each distribution and assume we have

$$p_\theta(\mathbf{x}|\mathbf{z}) = \mathcal{N}(\mathbf{x}; f(\mathbf{W}\mathbf{z}), \text{diag}(\boldsymbol{\sigma}_x^2)), \quad (3.2)$$

$$p_\theta(\mathbf{z}) = \mathcal{N}(\mathbf{z}; \boldsymbol{\mu}_z, \text{diag}(\boldsymbol{\sigma}_z^2)), \quad (3.3)$$

where f is an element-wise function (e.g., logistic sigmoid, tanh, or the identity), \mathbf{W} is a weight matrix, $\boldsymbol{\mu}_z$ is the prior mean, and $\boldsymbol{\sigma}_x^2$ and $\boldsymbol{\sigma}_z^2$ are vectors of variances.

In the simplest approach to inference, we can find the maximum a posteriori (MAP) estimate, that is, estimate the \mathbf{z}^* that maximizes $p_\theta(\mathbf{z}|\mathbf{x})$. While we cannot tractably evaluate $p_\theta(\mathbf{z}|\mathbf{x}) = \frac{p_\theta(\mathbf{x}, \mathbf{z})}{p_\theta(\mathbf{x})}$ directly, we can write

$$\begin{aligned} \mathbf{z}^* &= \arg \max_{\mathbf{z}} \frac{p_\theta(\mathbf{x}, \mathbf{z})}{p_\theta(\mathbf{x})} \\ &= \arg \max_{\mathbf{z}} p_\theta(\mathbf{x}, \mathbf{z}). \end{aligned}$$

Thus, we can perform inference using the joint distribution, $p_\theta(\mathbf{x}, \mathbf{z}) = p_\theta(\mathbf{x}|\mathbf{z})p_\theta(\mathbf{z})$, which we can tractably evaluate. We can also replace the optimization over the probability distribution with an optimization over the log probability, since $\log(\cdot)$ is a monotonically increasing function and does not affect the optimization. We then have

$$\begin{aligned} \mathbf{z}^* &= \arg \max_{\mathbf{z}} [\log p_\theta(\mathbf{x}|\mathbf{z}) + \log p_\theta(\mathbf{z})]. \\ &= \arg \max_{\mathbf{z}} [\log \mathcal{N}(\mathbf{x}; f(\mathbf{W}\mathbf{z}), \text{diag}(\boldsymbol{\sigma}_x^2)) + \log \mathcal{N}(\mathbf{z}; \boldsymbol{\mu}_z, \text{diag}(\boldsymbol{\sigma}_z^2))]. \end{aligned}$$

Each of the terms in this objective is a weighted squared error. For instance, the first term is the weighted squared error in reconstructing the data observation:

$$\begin{aligned} \log \mathcal{N}(\mathbf{x}; f(\mathbf{W}\mathbf{z}), \text{diag}(\boldsymbol{\sigma}_x^2)) &= \frac{-M}{2} \log(2\pi) - \frac{1}{2} \log \det(\text{diag}(\boldsymbol{\sigma}_x^2)) \\ &\quad - \frac{1}{2} \left\| \frac{\mathbf{x} - f(\mathbf{W}\mathbf{z})}{\boldsymbol{\sigma}_x} \right\|_2^2, \end{aligned}$$

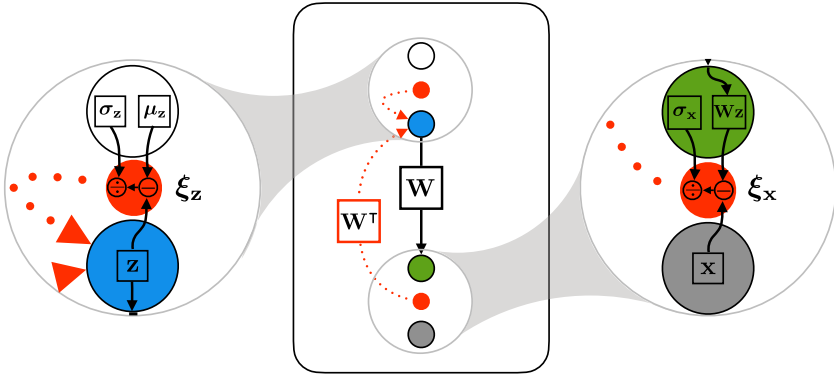


Figure 7: Hierarchical predictive coding. The diagram shows the basic computation graph for a gaussian latent variable model with MAP inference. The insets show the weighted error calculation for the latent (left) and observed (right) variables.

where M is the dimensionality of \mathbf{x} and $\|\cdot\|_2^2$ denotes the squared L2 norm. Plugging these terms into the objective and dropping terms that do not depend on \mathbf{z} yields

$$\mathbf{z}^* = \arg \max_{\mathbf{z}} \underbrace{\left[\frac{-1}{2} \left\| \frac{\mathbf{x} - f(\mathbf{Wz})}{\sigma_{\mathbf{x}}} \right\|_2^2 - \frac{1}{2} \left\| \frac{\mathbf{z} - \boldsymbol{\mu}_{\mathbf{z}}}{\sigma_{\mathbf{z}}} \right\|_2^2 \right]}_{\mathcal{L}(\mathbf{z}; \theta)}, \quad (3.4)$$

where we have defined the objective as $\mathcal{L}(\mathbf{z}; \theta)$. For purposes of illustration, let us assume that $f(\cdot)$ is the identity function, $f(\mathbf{Wz}) = \mathbf{Wz}$. We can then evaluate the gradient of $\mathcal{L}(\mathbf{z}; \theta)$ with regard to \mathbf{z} , yielding

$$\nabla_{\mathbf{z}} \mathcal{L}(\mathbf{z}; \theta) = \underbrace{\mathbf{W}^T \left(\frac{\mathbf{x} - \mathbf{Wz}}{\sigma_{\mathbf{x}}} \right)}_{\boldsymbol{\xi}_{\mathbf{x}}} - \underbrace{\frac{\mathbf{z} - \boldsymbol{\mu}_{\mathbf{z}}}{\sigma_{\mathbf{z}}}}_{\boldsymbol{\xi}_{\mathbf{z}}}. \quad (3.5)$$

The transposed weight matrix, \mathbf{W}^T , results from differentiating \mathbf{Wz} , translating the reconstruction error into an update in \mathbf{z} . We have also defined the weighted errors, $\boldsymbol{\xi}_{\mathbf{x}}$ and $\boldsymbol{\xi}_{\mathbf{z}}$. From equation 3.5, we see that if we want to perform inference using gradient-based optimization, such as $\mathbf{z} \leftarrow \mathbf{z} + \alpha \nabla_{\mathbf{z}} \mathcal{L}(\mathbf{z}; \theta)$, we need (1) the weighted errors, $\boldsymbol{\xi}_{\mathbf{x}}$ and $\boldsymbol{\xi}_{\mathbf{z}}$, and (2) the transposed weights, \mathbf{W}^T , or more generally, the Jacobian of the conditional likelihood mean. This overall scheme is depicted in Figure 7.

Table 1: Neural Correspondences of Hierarchical Predictive Coding.

Neuroscience	Predictive Coding
Top-down cortical projections	Generative model conditional mapping
Bottom-up cortical projections	Inference updating
Lateral inhibition	Covariance matrices
(Pyramidal) neuron activity	Latent variable estimates & errors
Cortical column	Corresponding estimate & error

To learn the weight parameters, we can differentiate $\mathcal{L}(\mathbf{z}; \theta)$ (see 3.4 with regard to) \mathbf{W} :

$$\nabla_{\mathbf{W}} \mathcal{L}(\mathbf{z}; \theta) = \xi_{\mathbf{x}} \mathbf{z}^{\top}.$$

This gradient is the product of a local error, $\xi_{\mathbf{x}}$, and the latent variable, \mathbf{z} , suggesting the possibility of a biologically plausible learning rule (Whittington & Bogacz, 2017).

Predictive coding identifies the conditional likelihood (equation 3.2) with backward (top-down) cortical projections, whereas inference (equation 3.5) is identified with forward (bottom-up) projections (Friston, 2005). Each is thought to be mediated by pyramidal neurons. Under this model, each cortical column predicts and estimates a stochastic continuous latent variable, possibly represented via a (pyramidal) firing rate or membrane potential (Friston, 2005). Interneurons within columns calculate errors ($\xi_{\mathbf{x}}$ and $\xi_{\mathbf{z}}$). Although we have only discussed diagonal covariance ($\sigma_{\mathbf{x}}^2$ and $\sigma_{\mathbf{z}}^2$), lateral inhibitory interneurons could parameterize full covariance matrices, $\Sigma_{\mathbf{x}}$ and $\Sigma_{\mathbf{z}}$, as a form of spatial predictive coding (see section 3.1). These factors weight $\xi_{\mathbf{x}}$ and $\xi_{\mathbf{z}}$, modulating the gain of each error as a form of “attention” (Feldman & Friston, 2010). Neural correspondences are summarized in Table 1.

We have presented a simplified model of hierarchical predictive coding, without multiple latent levels and dynamics. A full hierarchical predictive coding model would include these aspects and others. In particular, Friston has explored various design choices (Friston, Mattout, Trujillo-Barreto, Ashburner, & Penny, 2007; Friston, 2008a, 2008b), yet the core aspects of probabilistic generative modeling and variational inference remain the same. Elaborating and comparing these choices will be essential for empirically validating hierarchical predictive coding.

3.3 Empirical Support. While there is considerable evidence in support of predictions and errors in neural systems, disentangling these general aspects of predictive coding from the particular algorithmic choices remains challenging (Gershman, 2019). Here, we outline relevant work, but we

refer to Huang and Rao (2011), Bastos et al. (2012), Clark (2013), Keller and Mrsic-Flogel (2018), and Walsh, McGovern, Clark, and O'Connell (2020) for a more in-depth overview.

3.3.1 Spatiotemporal. Various works have investigated predictive coding in early sensory areas such as the retina (Srinivasan et al., 1982; Atick & Redlich, 1992). This involves fitting retinal ganglion cell responses to a spatial whitening (or decorrelation) process (Graham et al., 2006; Pitkow & Meister, 2012), which can be dynamically adjusted (Hosoya et al., 2005). Similar analyses suggest that retina also employs temporal predictive coding (Srinivasan et al., 1982; Palmer et al., 2015). Such models typically contain linear whitening filters (center-surround) followed by nonlinearities. These nonlinearities have been shown to be essential for modeling responses (Pitkow & Meister, 2012), possibly by inducing added sparsity (Graham et al., 2006). Spatiotemporal predictive coding also appears to be found in the thalamus (Dong & Atick, 1995; Dan et al., 1996) and cortex; however, such analyses are complicated by backward, modulatory inputs.

3.3.2 Hierarchical. Early work toward empirically validating hierarchical predictive coding came from explaining extraclassical receptive field effects (Rao & Ballard, 1999; Rao & Sejnowski, 2002), whereby top-down signals in the cortex alter classical visual receptive fields, suggesting that top-down influences play a key role in sensory processing (Gilbert & Sigman, 2007). Note that such effects support a cortical generative model generally (Olshausen & Field, 1997), not predictive coding specifically.

Temporal influences have been demonstrated through repetition suppression (Summerfield et al., 2006), in which activity diminishes in response to repeated (i.e., predictable) stimuli. This may reflect error suppression from improved predictions. Predictive coding has also been used to explain biphasic responses in LGN (Jehee & Ballard, 2009), in which reversing the visual input with an anticorrelated image results in a large response, presumably due to prediction errors. Predictive signals have been documented in auditory (Wacongne et al., 2011) and visual (Meyer & Olson, 2011) processing. Activity seemingly corresponding to prediction errors has also been observed in a variety of areas and contexts, including visual cortex in mice (Keller, Bonhoeffer, & Hübener, 2012; Zmarz & Keller, 2016; Gillon et al., 2021), auditory cortex in monkeys (Eliades & Wang, 2008) and rodents (Parras et al., 2017), and visual cortex in humans (Murray, Kersten, Olshausen, Schrater, & Woods, 2002; Alink, Schwiedrzik, Kohler, Singer, & Muckli, 2010; Egner, Monti, & Summerfield, 2010). Thus, sensory cortex appears to be engaged in hierarchical and temporal prediction, with prediction errors playing a key role.

Empirical evidence for predictive coding aside, given the complexity of neural systems, the theory is undoubtedly incomplete or incorrect. Without the low-level details such as connectivity and potentials, it is difficult

to determine the computational form of the circuit. Further, these models are typically oversimplified, with few trained parameters, detached from natural stimuli. While new tools enable us to test predictive coding in neural circuits (Gillon et al., 2021), machine learning, particularly VAEs, can advance from the other direction. Training large-scale models on natural stimuli may improve empirical predictions for biological systems (Rao & Ballard, 1999; Lotter et al., 2018).

4 Variational Autoencoders

Variational autoencoders (VAEs) (Kingma & Welling, 2014; Rezende et al., 2014) are latent variable models parameterized by deep networks. As in hierarchical predictive coding, these models typically contain gaussian latent variables and are trained using variational inference. However, rather than performing inference optimization directly, VAEs amortize inference (Gershman & Goodman, 2014).

4.1 Amortized Variational Inference. Amortization refers to spreading out costs. In amortized inference, these are the computational costs of inference. With $q(\mathbf{z}|\mathbf{x}) = \mathcal{N}(\mathbf{z}; \boldsymbol{\mu}_q, \text{diag}(\boldsymbol{\sigma}_q^2))$ and $\boldsymbol{\lambda} \equiv [\boldsymbol{\mu}_q, \boldsymbol{\sigma}_q]$, rather than separately optimizing $\boldsymbol{\lambda}$ for each data example, we amortize this optimization using a learned optimizer or inference model. By using meta-optimization, we can perform inference optimization far more efficiently. Inference models are linked with deep latent variable models, popularized by the Helmholtz machine (Dayan et al., 1995), a form of autoencoder (Ballard, 1987). Here, the inference model is a direct mapping from \mathbf{x} to $\boldsymbol{\lambda}$,

$$\boldsymbol{\lambda} \leftarrow f_\phi(\mathbf{x}), \quad (4.1)$$

where f_ϕ is a function (deep network) with parameters ϕ . We then denote the approximate posterior as $q_\phi(\mathbf{z}|\mathbf{x})$ to denote the parameterization by ϕ . Now, rather than optimizing $\boldsymbol{\lambda}$ using gradient-based techniques, we update ϕ using $\nabla_\phi \mathcal{L} = \frac{\partial \mathcal{L}}{\partial \boldsymbol{\lambda}} \frac{\partial \boldsymbol{\lambda}}{\partial \phi}$, thereby letting f_ϕ learn to optimize $\boldsymbol{\lambda}$. This procedure is simple, as we only need to tune the learning rate for ϕ , and efficient, as we have an estimate of $\boldsymbol{\lambda}$ after one forward pass through f_ϕ . Amortization is also widely applicable: if we can estimate $\nabla_{\boldsymbol{\lambda}} \mathcal{L}$, we can continue differentiating through the chain $\phi \rightarrow \boldsymbol{\lambda} \rightarrow \mathbf{z} \rightarrow \mathcal{L}$.

To differentiate through $\mathbf{z} \sim q_\phi(\mathbf{z}|\mathbf{x})$, we can use the pathwise derivative estimator, also referred to as the reparameterization estimator (Kingma & Welling, 2014). This is accomplished by expressing \mathbf{z} in terms of an auxiliary random variable. The most common example expresses $\mathbf{z} \sim \mathcal{N}(\mathbf{z}; \boldsymbol{\mu}_q, \text{diag}(\boldsymbol{\sigma}_q^2))$ as $\mathbf{z} = \boldsymbol{\mu}_q + \boldsymbol{\epsilon} \odot \boldsymbol{\sigma}_q$, where $\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{\epsilon}; \mathbf{0}, \mathbf{I})$ and \odot denotes element-wise multiplication. We can then estimate $\nabla_{\boldsymbol{\mu}_q} \mathcal{L}$ and $\nabla_{\boldsymbol{\sigma}_q} \mathcal{L}$, allowing us to calculate the inference model gradients, $\nabla_\phi \mathcal{L}$.

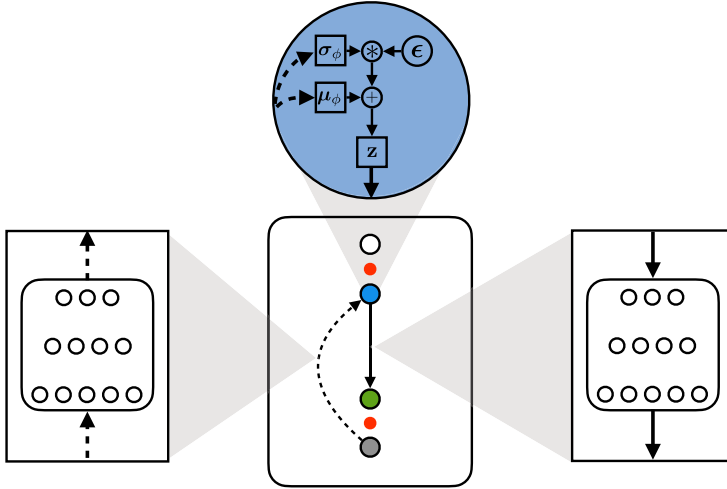


Figure 8: Variational autoencoder. VAEs use direct amortization (see equation 4.1) to train deep latent variable models. The inference model (left) is an encoder, and the conditional latent likelihood (right) is a decoder. Each is parameterized by deep networks.

When direct amortization is combined with the pathwise derivative estimator in deep latent variable models, the resulting setup is a variational autoencoder (Kingma & Welling, 2014; Rezende et al., 2014). In this interpretation, $q_\phi(\mathbf{z}|\mathbf{x})$ is an encoder, \mathbf{z} is the latent code, and $p_\theta(\mathbf{x}|\mathbf{z})$ is a decoder (see Figure 8). This direct encoding scheme is intuitive: in the same way that $p_\theta(\mathbf{x}|\mathbf{z})$ directly maps \mathbf{z} to a distribution over \mathbf{x} , $q_\phi(\mathbf{z}|\mathbf{x})$ directly maps \mathbf{x} to a distribution over \mathbf{z} . Indeed, with perfect knowledge of $p_\theta(\mathbf{x}, \mathbf{z})$, f_ϕ could act as a lookup table, mapping each \mathbf{x} to the corresponding optimal λ . However, in practice, direct amortization of this form tends to result in suboptimal estimates of λ (Cremer, Li, & Duvenaud, 2018), motivating more powerful amortized inference techniques.

4.1.1 Iterative Amortized Inference. One method for improving direct amortization involves incorporating iterative updates (Hjelm et al., 2016; Krishnan et al., 2018; Kim, Wiseman, Miller, Sontag, & Rush, 2018; Marino, Yue, & Mandt, 2018), replacing a one-step inference procedure with a multistep procedure. Iterative amortized inference (Marino, Yue, et al., 2018) maintains an inference model, but uses it to perform iterative updates on the approximate posterior. Following the previous notation, the basic form of an iterative amortized inference model is given as

$$\lambda \leftarrow f_\phi(\lambda, \nabla_\lambda \mathcal{L}). \quad (4.2)$$

Iterative inference models take in the current estimate, λ , as well as the gradient, $\nabla_{\lambda}\mathcal{L}$, and output an updated estimate of λ . As before, inference model parameters are updated using estimates of $\nabla_{\phi}\mathcal{L}$. Note that equation 4.2 generalizes stochastic gradient-based optimization. For instance, a special case is $\lambda \leftarrow \lambda + \alpha \nabla_{\lambda}\mathcal{L}$, where α is a step-size; however, equation 4.2 also includes nonlinear updates (Andrychowicz et al., 2016).

In latent gaussian models, ∇_{λ} is defined by the weighted errors, ξ_x and ξ_z , and the Jacobian of the conditional likelihood, J (\mathbf{W} in the linear model in section 3.2). Thus, in latent gaussian models, we can consider inference models of the special form

$$\lambda \leftarrow f_{\phi}(\lambda, \xi_x, \xi_z). \quad (4.3)$$

This is a learned, nonlinear mapping from errors to updated estimates of the approximate posterior, that is, learned negative feedback. (The distinction between direct and iterative amortization is shown in Figures 10b and 10c.) Iterative amortization can be readily extended to sequential models (Marino, Cvitkovic, & Yue, 2018), resulting in a general predict-update inference scheme, highly reminiscent of Kalman filtering (Kalman, 1960).

4.2 Extensions of VAEs.

4.2.1 Additional Dependencies and Representation Learning. VAEs have been extended to a variety of architectures, incorporating hierarchical and temporal dependencies. Sønderby et al. (2016) proposed a hierarchical VAE, in which the conditional prior at each level, ℓ , that is, $p_{\theta}(\mathbf{z}^{\ell}|\mathbf{z}^{\ell+1:L})$, is parameterized by a deep network (see equation 2.7). Follow-up works have scaled this approach with impressive results (Kingma et al., 2016; Vahdat & Kautz, 2020; Child, 2020), extracting increasingly abstract features at higher levels (Maaløe, Fraccaro, Lievin, & Winther, 2019). Another line of work has incorporated temporal dependencies within VAEs, parameterizing dynamics in the prior and conditional likelihood with deep networks (Chung et al., 2015; Fraccaro, Sønderby, Paquet, & Winther, 2016). Such models can also provide representations and predictions for reinforcement learning (Ha & Schmidhuber, 2018; Hafner et al., 2019).

Other work has investigated representation learning within VAEs. One approach, the β -VAE (Higgins et al., 2017), modifies the ELBO (see equation 2.14) by adjusting a weighting, β , on $D_{\text{KL}}(q(\mathbf{z}|\mathbf{x})||p_{\theta}(\mathbf{z}))$. This tends to yield more disentangled (i.e., independent) latent variables. Indeed, β controls the rate-distortion trade-off between latent complexity and reconstruction (Alemi et al., 2018), highlighting VAEs' ability to extract latent structure at multiple resolutions (Rezende & Viola, 2018). A separate line of work has focused on identifiability: the ability to uniquely recover the original latent variables within a model (or their posterior). While this is true in linear ICA

(Comon, 1994), it is not generally the case with nonlinear ICA and noninvertible models (VAEs) (Khemakhem, Kingma, Monti, & Hyvarinen, 2020; Gresele, Fissore, Javaloy, Schölkopf, & Hyvarinen, 2020), requiring special considerations.

4.2.2 Normalizing Flows. Another direction within VAEs is the use of normalizing flows (Rezende & Mohamed, 2015). Flow-based distributions use invertible transforms to add and remove dependencies (see section 2.2.1). While such models can operate as generative models (Dinh et al., 2015, 2017; Papamakarios, Pavlakou, & Murray, 2017), they can also define distributions in VAEs. This includes the approximate posterior (Rezende & Mohamed, 2015), prior (Huang et al., 2017), and conditional likelihood (Agrawal & Dukkipati, 2016). In each case, a deep network outputs the parameters (e.g., mean and variance) of a base distribution over a normalized variable. Separate deep networks parameterize the transforms, which map between the normalized and unnormalized variables.

4.2.3 Example. Consider a normalized variable, \mathbf{u} , defined by the distribution $p_\theta(\mathbf{u}|\cdot) = \mathcal{N}(\mathbf{u}; \boldsymbol{\mu}_\theta(\cdot), \text{diag}(\boldsymbol{\sigma}_\theta^2(\cdot)))$, where $\boldsymbol{\mu}_\theta$ and $\boldsymbol{\sigma}_\theta$ are output by deep networks, with \cdot denoting conditioning input variables. We consider an affine transform (Dinh et al., 2017), defined by a shift vector, $\boldsymbol{\alpha} = \boldsymbol{\alpha}_\theta(\mathbf{u})$, and a scale matrix, $\mathbf{B} = \mathbf{B}_\theta(\mathbf{u})$, each of which may be parameterized by deep networks. This defines a new, unnormalized variable \mathbf{v} ,

$$\mathbf{v} = \boldsymbol{\alpha} + \mathbf{B}\mathbf{u}, \quad (4.4)$$

which can now contain affine dependencies between dimensions. For equation 4.4 to be invertible, we require \mathbf{B} itself that is, to be invertible, that is, nonzero determinant. Thus, \mathbf{B} is a square matrix, and \mathbf{u} and \mathbf{v} are the same dimensionality. If instead we are given \mathbf{v} , we can calculate its log probability by applying the normalizing inverse transform to get \mathbf{u} :

$$\mathbf{u} = \mathbf{B}^{-1}(\mathbf{v} - \boldsymbol{\alpha}), \quad (4.5)$$

then use the change-of-variables formula, equation 2.6. This converts the log-probability calculation from the structured space of \mathbf{v} to the normalized space of \mathbf{u} . Note that the multivariate gaussian density, equation 2.9, is a special case of this transform, taking a standard gaussian variable, $\mathbf{u} \sim \mathcal{N}(\mathbf{u}; \mathbf{0}, \mathbf{I})$, and adding linear dependencies to yield a multivariate gaussian variable, $\mathbf{v} \sim \mathcal{N}(\mathbf{v}; \boldsymbol{\alpha}, \mathbf{B}^\top \mathbf{B})$. In this case, where $\boldsymbol{\alpha}$ and \mathbf{B} are constant, the inverse transform removes linear dependencies between dimensions in \mathbf{v} . We depict this scheme in Figure 9, along with the more general nonlinear version provided by normalizing flows, in which $\boldsymbol{\alpha}_\theta$ and \mathbf{B}_θ are functions. Thus, normalizing flows provides a powerful, more general approach for

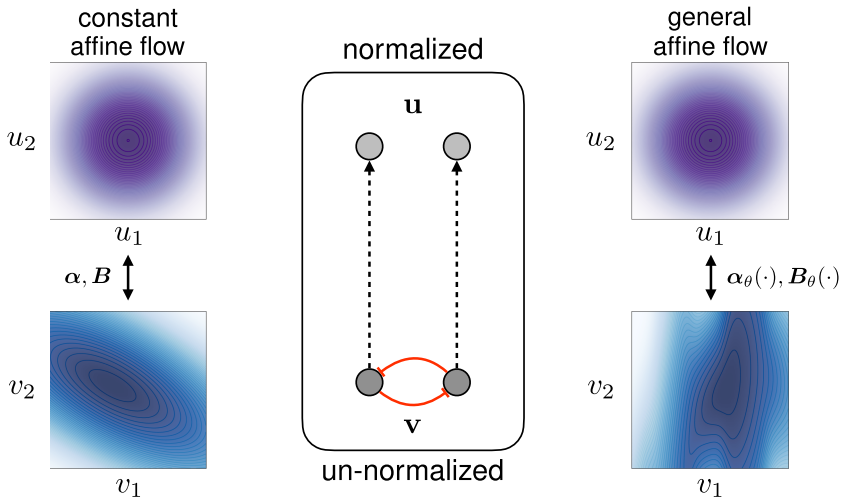


Figure 9: Normalizing flows. Normalizing flows is a framework for adding or removing dependencies. Using affine parameter functions, α_θ and B_θ , one can model nonlinear dependencies, generalizing constant transforms, e.g., a covariance matrix.

augmenting the distributions in latent variable models, applicable across both space (Rezende & Mohamed, 2015; Kingma et al., 2016) and time (van den Oord et al., 2018; Marino, Chen, He, & Mandt, 2020).

5 Connections and Comparisons

Predictive coding and VAEs (and deep generative models generally), are highly related in both their model formulations and inference approaches (see Figure 10). Specifically,

- **Model formulation:** Both areas consider hierarchical latent gaussian models with nonlinear dependencies between latent levels, as well as dependencies within levels via covariance matrices (predictive coding) or normalizing flows (VAEs).
- **Inference:** Both areas use variational inference, often with gaussian approximate posteriors. While predictive coding and VAEs employ differing optimization techniques, these are design choices in solving the same inference problem.

These similarities reflect a common mathematical foundation inherited from cybernetics and descendant fields. We now discuss these two points in more detail.

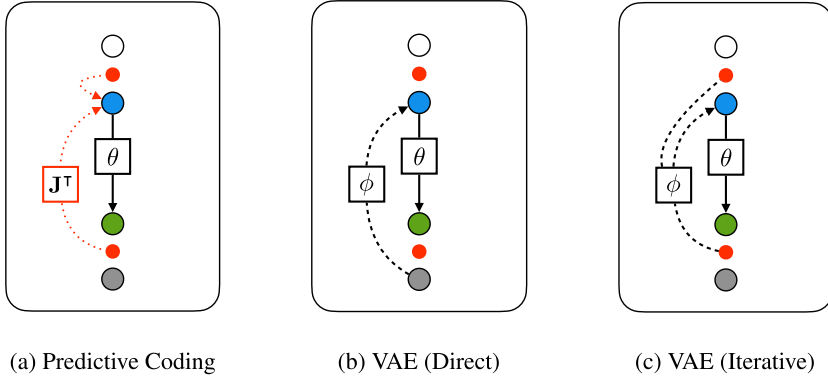


Figure 10: Hierarchical predictive coding and VAEs. Computation diagrams for (a) hierarchical predictive coding, (b) VAE with direct amortized inference, and (c) VAE with iterative amortized inference (Marino, Yue, et al., 2018). J^\top denotes the transposed Jacobian matrix of the conditional likelihood. Red dotted lines denote gradients, and black dashed lines denote amortized inference.

5.1 Model Formulation. The primary distinction in model formulation is the form of the (non-linear) functions parameterizing dependencies. Rao and Ballard (1999) parameterize the conditional likelihood as a linear function followed by an element-wise nonlinearity. Friston has considered a wider range of functions, such as polynomial (Friston, 2008a); however, such functions are rarely learned. VAEs instead parameterize these functions using deep networks with multiple layers. The deep network weights are trained through backpropagation, enabling the wide application of VAEs to various data domains.

Predictive coding and VAEs also consider dependencies within each level. Friston (2005) uses full-covariance gaussian densities, with the inverse of the covariance matrix (precision) parameterizing linear dependencies within a level. Rao and Ballard (1999) normalize the observations, modeling linear dependencies within the conditional likelihood. These are linear special cases of the more general technique of normalizing flows (Rezende & Mohamed, 2015): a covariance matrix is an affine normalizing flow with linear dependencies (Kingma et al., 2016). Normalizing flows have been applied throughout each of the distributions within VAEs (see section 4.2.2), modeling nonlinear dependencies across both spatial and temporal dimensions. These flows are also parameterized by deep networks, providing a flexible yet general modeling approach.

Related to normalization, there are proposals within predictive coding that the precision of the prior could mediate a form of attention (Feldman & Friston, 2010). Increasing the precision of a variable serves as a form of gain modulation, up-weighting the error in the objective function, thereby

enforcing more accurate inference estimates. This concept is absent from VAEs. However, as VAEs become more prevalent in interactive settings (Ha & Schmidhuber, 2018), that is, beyond pure generative modeling, this may become crucial in steering models toward task-relevant perceptual inferences.

Finally, predictive coding and VAEs have both been extended to sequential settings. In predictive coding, sequential dependencies may be parameterized by linear functions (Srinivasan et al., 1982) or so-called generalized coordinates (Friston, 2008a), modeling multiple orders of motion. In extensions of VAEs, sequential dependencies are again parameterized by deep networks, in many cases using recurrent networks (Chung et al., 2015; Fracaro et al., 2016). Thus, while the specific implementations vary, in either case, sequential dependencies are ultimately functions, which are subject to design choices.

5.2 Inference. Although both predictive coding and VAEs typically use variational inference with gaussian approximate posteriors, sections 3 and 4 illustrate key differences (see Figure 10). Predictive coding generally relies on gradient-based optimization to perform inference, whereas VAEs employ amortized optimization. While these approaches may at first appear radically different, hybrid error-encoding inference approaches (see equation 4.3), such as PredNet (Lotter et al., 2017) and iterative amortization (Marino, Yue, et al., 2018), provide a link. Such approaches receive errors as input, as in predictive coding; however, they have learnable parameters (i.e., amortization). In fact, amortization may provide a crucial element for implementing predictive coding in biological neural networks.

Though rarely discussed, hierarchical predictive coding assumes that the inference gradients, supplied by forward connections, can be readily calculated. But as seen in section 3.2, the weights of these forward connections are the transposed Jacobian matrix of the backward connections (Rao & Ballard, 1999). This is an example of the weight transport problem (Grossberg, 1987), in which the weights of one set of connections (forward) depend on the weights of another set of connections (backward). This is generally regarded as not being biologically plausible.

Amortization provides a solution to this problem: *learn* to perform inference. Rather than transporting the generative weights to the inference connections, amortization learns a separate set of inference weights, potentially using local learning rules (Bengio, 2014; Lee, Zhang, Fischer, & Bengio, 2015). Thus, despite criticism from Friston (2018), amortization may offer a more biologically plausible inference approach. Further, amortized inference yields accurate estimates with exceedingly few iterations: even a single iteration may yield reasonable estimates (Marino, Yue, et al., 2018). These computational efficiency benefits provide another argument in favor of amortization.

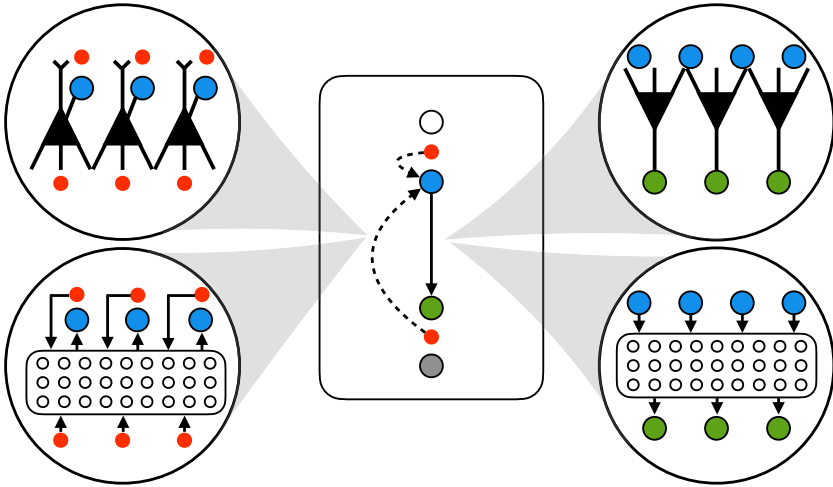


Figure 11: Pyramidal neurons and deep networks. Connecting VAEs with predictive coding places deep networks (bottom) in correspondence with the dendrites of pyramidal neurons (top), for both generation (right) and (amortized) inference (left).

Finally, although predictive coding and VAEs typically assume gaussian approximate posteriors, there is one additional difference in the ways in which these parameters are conventionally calculated. Friston often uses the Laplace approximation³ (Friston et al., 2007), solving directly for the optimal gaussian variance, whereas VAEs treat this as another output of the inference model (Kingma & Welling, 2014; Rezende, Mohamed, & Wierstra, 2014). These approaches can be applied in either setting (Park, Kim, & Kim, 2019).

6 Correspondences

Having connected VAEs and predictive coding, we now discuss possible correspondences between machine learning and neuroscience. In Table 1, top-down and bottom-up cortical projections, each mediated by pyramidal neurons, respectively parameterize the generative model and inference updates. Mapping this onto VAEs suggests that deep (artificial) neural networks are in correspondence with pyramidal neuron dendrites (see Figure 11, right), or, more specifically, a deep network corresponds to a

³This is not to be confused with a Laplace distribution. The approximate posterior is still gaussian.

collection of pyramidal dendrites operating in parallel. Predictive coding also postulates that lateral interneurons parameterize dependencies within variables as inverse covariance matrices. VAEs parameterize these dependencies using more general normalizing flows, suggesting that normalizing flows are in correspondence with lateral interneurons. While normalizing flows also use deep networks, the effect that they have on computation tends to be restricted and simple (e.g., affine). We now briefly discuss these correspondences.

6.1 Pyramidal Neurons and Deep Networks.

6.1.1 Nonlinear Dendritic Computation. Placing deep networks in correspondence with pyramidal dendrites suggests that (some) biological neurons may be better computationally described as nonlinear functions. Evidence from neuroscience supports this claim. Early simulations showed that individual pyramidal neurons, through dendritic processing, could operate as multilayer artificial networks (Zador, Claiborne, & Brown, 1992; Mel, 1992). This was later supported by empirical findings that pyramidal dendrites act as computational “subunits,” yielding the equivalent of a two-layer artificial network (Poirazi, Brannon, & Mel, 2003; Polsky, Mel, & Schiller, 2004). More recently, Gidon et al. (2020) demonstrated that individual pyramidal neurons can compute the XOR operation, which requires nonlinear processing. This is supported by further modeling work (Jones & Kording, 2020; Beniaguev, Segev, & London, 2021). Positing a more substantial role for dendritic computation (London & Häusser, 2005) moves beyond the simplistic comparison of biological and artificial neurons that currently dominates. Instead, neural computation depends on morphology and circuits.

6.1.2 Amortization. Pyramidal neurons mediate both top-down and bottom-up cortical projections. Under predictive coding, this suggests that inference relies on learned, nonlinear functions: amortization. One such implementation is through pyramidal neurons with separate apical and basal dendrites, which, respectively, receive top-down and bottom-up inputs (Bekkers, 2011; Guergiev, Lillicrap, & Richards, 2016). Recent evidence from Gillon et al. (2021) suggests that these are top-down predictions and bottom-up errors. These neurons may implement iterative amortized inference (Marino, Yue, et al., 2018), separately processing top-down and bottom-up error signals to update inference estimates (see Figure 11, left). While some empirical support for amortization exists (Yildirim, Kulikarni, Freiwald, & Tenenbaum, 2015; Dasgupta, Schulz, Goodman, & Gershman, 2018), further investigation is needed. Finally, this perspective implies separate computational processing for prediction and inference, with distinct (but linked) frequencies. While some evidence supports this conjecture

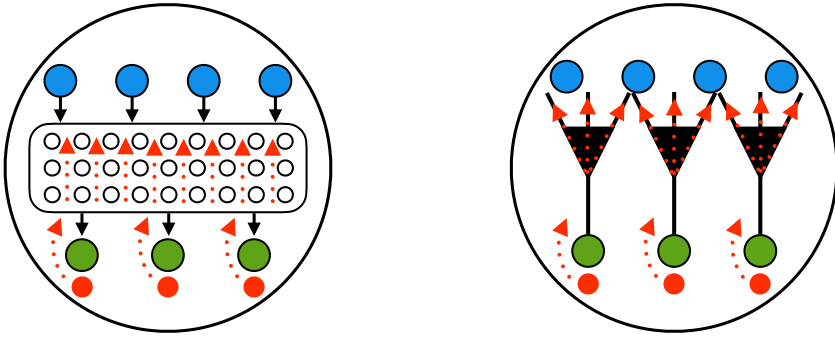


Figure 12: Backpropagation within neurons. If deep networks are in correspondence with pyramidal neurons, this implies that backpropagation (left) is analogous to learning within neurons, perhaps via backpropagating action potentials (right).

(Bastos et al., 2015), it is unclear how this could be implemented in biological neurons.

6.1.3 Backpropagation. The biological plausibility of backpropagation is an open question (Lillicrap, Santoro, Marris, Akerman, & Hinton, 2020). Critics argue that backpropagation requires nonlocal learning signals (Grossberg, 1987; Crick, 1989), whereas the brain relies largely on local learning rules (Hebb, 1949; Markram, Lübke, Frotscher, & Sakmann, 1997; Bi & Poo, 1998). Biologically plausible formulations of backpropagation have been proposed (Stork, 1989; Körding & König, 2001; Xie & Seung, 2003; Hinton, 2007; Lillicrap, Cownden, Tweed, & Akerman, 2016), attempting to reconcile this disparity. Yet consensus is still lacking. From another perspective, the apparent biological implausibility of backpropagation may instead be the result of incorrectly assuming a one-to-one correspondence between biological and artificial neurons.

Placing deep networks in correspondence with pyramidal neurons suggests a different perspective on the biological plausibility debate. In hierarchical latent variable models, prediction errors at each level provide a local learning signal (Friston, 2005; Bengio, 2014; Lee et al., 2015; Whittington & Bogacz, 2017). Thus, learning within each latent level is performed through optimization of local errors. This is exemplified by hierarchical VAEs (Sønderby, Raiko, Maaløe, Sønderby, & Winther, 2016), which utilize backpropagation within each latent level but not across levels. This suggests that learning within pyramidal neurons may be more analogous to backpropagation (see Figure 12). One possible candidate is backpropagating action potentials (Stuart & Sakmann, 1994; Williams & Stuart,

2000), which propagate a signal of neural activity back to synaptic inputs (Stuart, Spruston, Sakmann, & Häusser, 1997; Brunner & Szabadics, 2016), resulting in a variety of synaptic changes throughout the dendrites (Johanning et al., 2015). While computational models from Schiess, Urbanczik, and Senn (2016) support this conjecture, further investigation is needed.

6.2 Lateral Inhibition and Normalizing Flows.

6.2.1 Sensory Input Normalization. One of the key computational roles of early sensory areas appears to be reducing spatiotemporal redundancies—normalization. In retina, this is performed through lateral inhibition via horizontal and amacrine cells, removing correlations (Graham et al., 2006; Pitkow & Meister, 2012). Normalization and prediction are inseparable, and, accordingly, previous work has framed early sensory processing in terms of spatiotemporal predictive coding (Srinivasan et al., 1982; Hosoya et al., 2005; Palmer et al., 2015). This is often motivated in terms of increased sensitivity or efficiency (Srinivasan et al., 1982; Atick & Redlich, 1990) due to redundancy reduction (Barlow, 1961a; Barlow et al., 1989), that is, compression.

If we consider cortex as a hierarchical latent variable model, then early sensory areas are implicated in parameterizing the conditional likelihood. The ubiquity of normalization in these areas is suggestive of normalization in a flow-based model, implementing the inference direction of a flow-based conditional likelihood (Agrawal & Dukkipati, 2016; Winkler et al., 2019). In addition to the sensitivity and efficiency benefits cited above, this learned, normalized space simplifies downstream generative modeling and improves generalization (Marino et al., 2020).

6.2.2 Normalization in Thalamus. Normalization also appears to occur in first-order thalamic relays, such as the lateral geniculate nucleus (LGN). Dong and Atick (1995) framed LGN in terms of temporal normalization, with supporting evidence provided by Dan et al. (1996). This has the effect of removing predictable temporal structure (e.g., static backgrounds). Under the interpretation above, this is an additional inference stage of a flow-based conditional likelihood (Marino et al., 2020).

6.2.3 Normalization in Cortex. Normalization, via local lateral inhibition, is also found throughout cortex (King et al., 2013). Friston (2005) suggested that this plays the role of inverse covariance (precision) matrices, modeling dependencies between dimensions within the same latent level of the hierarchy. This corresponds to parameterizing approximate posteriors (Rezende & Mohamed, 2015; Kingma et al., 2016) and/or conditional priors (Huang et al., 2017) with affine normalizing flows with linear

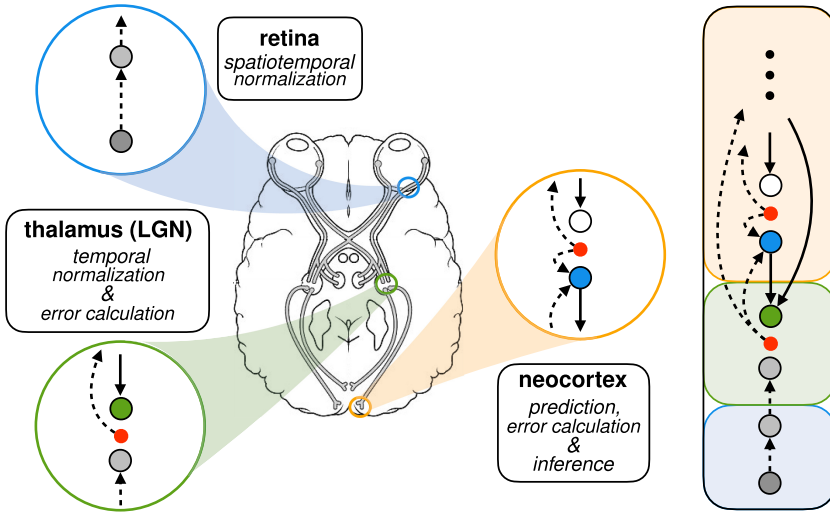


Figure 13: Visual pathway. The retina and LGN are interpreted as implementing normalizing flows, that is, spatiotemporal predictive coding, reducing spatial and temporal redundancy in the visual input (dashed arrows between gray circles). LGN is also the lowest level for hierarchical predictions from cortex. Using prediction errors throughout the hierarchy, forward cortical connections update latent estimates.

dependencies.⁴ Normalizing flows offers a more general framework for describing these computations. Further, Friston (2005) assumes that these dependencies are modeled using *symmetric* weights, whereas normalizing flows permits *non-symmetric* schemes, e.g., using autoregressive models (Kingma et al., 2016) or ensembles (Uria et al., 2014). These weights can also be restricted to local spatial regions (Vahdat & Kautz, 2020). Similar normalization operations can also parameterize temporal dynamics (Marino et al., 2020), akin to Friston’s generalized coordinates (Friston, 2008a). The overall computational scheme is shown in Figure 13.

7 Discussion

We have reviewed predictive coding and VAEs, identifying their shared history and formulations. These connections provide an invaluable link between leading areas of theoretical neuroscience and machine learning, hopefully facilitating the transfer of ideas across fields. We have initiated

⁴Specifically, Friston (2005) employs zero-phase component analysis (ZCA) whitening, whereas Kingma et al. (2016) explored Cholesky whitening.

this process by proposing two novel correspondences suggested by this perspective: (1) dendrites of pyramidal neurons and deep networks and (2) lateral inhibition and normalizing flows. Placing pyramidal neurons in correspondence with deep networks departs from the traditional one-to-one analogy of biological and artificial neurons, raising questions regarding dendritic computation and backpropagation. Normalizing flows offers a more general framework for normalization via lateral inhibition. Connecting these areas may provide new insights for both machine learning and neuroscience, helping us move beyond overly simplistic comparisons.

7.1 Predictive Coding → VAEs. Although considerable independent progress has recently occurred in VAEs, such models are often still trained on relatively simple, standardized data sets of static images. Thus, predictive coding and neuroscience may still hold insights for improving these models for real-world settings. For instance, the correspondences outlined above may offer new architectural insights in designing deep networks and normalizing flows, for example, drawing on dendritic morphology, short-term plasticity, and connectivity. Predictive coding has also used prediction precision as a form of attention (Feldman & Friston, 2010). More broadly, neuroscience may provide insights into interfacing VAEs with other computations, as well as within embodied agents.

7.2 VAEs → Predictive Coding. Another motivating factor in connecting these areas stems from a desire for large-scale, testable models of predictive coding. While predictive coding offers general considerations for neural activity, e.g., predictions, prediction errors, and extra-classical receptive fields (Rao & Ballard, 1999), it is difficult to align such hypotheses with real data due to the many possible design choices (Gershman, 2019). Current models are often implemented in simplified settings, with few, if any, learned parameters. VAEs, in contrast, offer a large-scale test-bed for implementing models and evaluating them on natural stimuli. This may offer a more nuanced perspective over current efforts to compare biological and artificial neural activity (Yamins et al., 2014).

While we have reviewed many topics across neuroscience and machine learning, for brevity, we have focused exclusively on passive perceptual settings. However, separate, growing bodies of work are incorporating predictive coding (Adams, Shipp, & Friston, 2013) and VAEs (Ha & Schmidhuber, 2018) within active settings such as reinforcement learning. We are hopeful that the connections in this paper will inspire further insight in such areas.

Appendix A: Variational Bound Derivation

We can express the KL divergence between $q(\mathbf{z}|\mathbf{x})$ and $p_\theta(\mathbf{z}|\mathbf{x})$ as

$$D_{\text{KL}}(q(\mathbf{z}|\mathbf{x})||p_\theta(\mathbf{z}|\mathbf{x})) = \mathbb{E}_{\mathbf{z}\sim q(\mathbf{z}|\mathbf{x})} [\log q(\mathbf{z}|\mathbf{x}) - \log p_\theta(\mathbf{z}|\mathbf{x})]$$

$$\begin{aligned}
&= \mathbb{E}_{\mathbf{z} \sim q(\mathbf{z}|\mathbf{x})} \left[\log q(\mathbf{z}|\mathbf{x}) - \log \left(\frac{p_{\theta}(\mathbf{x}, \mathbf{z})}{p_{\theta}(\mathbf{x})} \right) \right] \\
&= \underbrace{\mathbb{E}_{\mathbf{z} \sim q(\mathbf{z}|\mathbf{x})} [\log q(\mathbf{z}|\mathbf{x}) - \log p_{\theta}(\mathbf{x}, \mathbf{z})]}_{-\mathcal{L}(\mathbf{x}; q, \theta)} + \log p_{\theta}(\mathbf{x}). \quad (\text{A.1})
\end{aligned}$$

Acknowledgments

Sam Gershman and Rajesh Rao provided helpful comments on this manuscript, and Karl Friston engaged in useful early discussions related to these ideas. We also thank the anonymous reviewers for their feedback and suggestions.

References

- Ackley, D. H., Hinton, G. E., & Sejnowski, T. J. (1985). A learning algorithm for Boltzmann machines. *Cognitive Science*, 9(1), 147–169. 10.1207/s15516709cog0901_7
- Adams, R. A., Shipp, S., & Friston, K. J. (2013). Predictions not commands: Active inference in the motor system. *Brain Structure and Function*, 218(3), 611–643. 10.1007/s00429-012-0475-5, PubMed: 23129312
- Agrawal, S., & Dukkipati, A. (2016). *Deep variational inference without pixel-wise reconstruction*. arXiv:1611.05209.
- Alemi, A., Poole, B., Fischer, I., Dillon, J., Sauro, R. A., & Murphy, K. (2018). Fixing a broken ELBO. In *Proceedings of the International Conference on Machine Learning* (pp. 159–168). New York: ACM.
- Alink, A., Schwiedrzik, C. M., Kohler, A., Singer, W., & Muckli, L. (2010). Stimulus predictability reduces responses in primary visual cortex. *Journal of Neuroscience*, 30(8), 2960–2966. 10.1523/JNEUROSCI.3730-10.2010, PubMed: 20181593
- Andrychowicz, M., Denil, M., Gomez, S., Hoffman, M. W., Pfau, D., Schaul, T., & de Freitas, N. (2016). Learning to learn by gradient descent by gradient descent. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, & R. Garnett (Eds.), *Advances in neural information processing systems*, 29 (pp. 3981–3989). Red Hook, NY: Curran.
- Ashby, W. R. (1956). *An introduction to cybernetics*. London: Chapman and Hall.
- Atal, B., & Schroeder, M. (1979). Predictive coding of speech signals and subjective error criteria. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 27(3), 247–254. 10.1109/TASSP.1979.1163237
- Atick, J. J., & Redlich, A. N. (1990). Towards a theory of early visual processing. *Neural Computation*, 2(3), 308–320. 10.1162/neco.1990.2.3.308
- Atick, J. J., & Redlich, A. N. (1992). What does the retina know about natural scenes? *Neural Computation*, 4(2), 196–210. 10.1162/neco.1992.4.2.196
- Baccus, S. A., Ölveczky, B. P., Manu, M., & Meister, M. (2008). A retinal circuit that computes object motion. *Journal of Neuroscience*, 28(27), 6807–6817. 10.1523/JNEUROSCI.4206-07.2008, PubMed: 18596156
- Ballard, D. H. (1987). Modular learning in neural networks. In *Proceedings of the AAAI* (pp. 279–284).

- Barlow, H. B. (1961a). Possible principles underlying the transformation of sensory messages. *Sensory Communication*, 1, 217–234.
- Barlow, H. B. (1961b). The coding of sensory messages. *Current problems in animal behavior*. Cambridge: Cambridge University Press.
- Barlow, H. B. (1989). Unsupervised learning. *Neural Computation*, 1(3), 295–311. 10.1162/neco.1989.1.3.295
- Barlow, H. B., Kaushal, T. P., & Mitchison, G. J. (1989). Finding minimum entropy codes. *Neural Computation*, 1(3), 412–423. 10.1162/neco.1989.1.3.412
- Bastos, A. M., Usrey, W. M., Adams, R. A., Mangun, G. R., Fries, P., & Friston, K. (2012). Canonical microcircuits for predictive coding. *Neuron*, 76(4), 695–711. 10.1016/j.neuron.2012.10.038, PubMed: 23177956
- Bastos, A. M., Vezoli, J., Bosman, C. A., Schoffelen, J.-M., Oostenveld, R., Dowdall, J. R., . . . Fries, P. (2015). Visual areas exert feedforward and feedback influences through distinct frequency channels. *Neuron*, 85(2), 390–401. 10.1016/j.neuron.2014.12.018, PubMed: 25556836
- Bekkers, J. M. (2011). Pyramidal neurons. *Current Biology*, 21(24), R975. 10.1016/j.cub.2011.10.037, PubMed: 22192824
- Bell, A. J., & Sejnowski, T. J. (1997). The “independent components” of natural scenes are edge filters. *Vision Research*, 37(23), 3327–3338. 10.1016/S0042-6989(97)00121-1, PubMed: 9425547
- Bengio, Y. (2014). *How autoencoders could provide credit assignment in deep networks via target propagation*. arXiv:1407.7906.
- Bengio, Y., & Bengio, S. (2000). Modeling high-dimensional discrete data with multilayer neural networks. In S. Solla, T. Leen, & K. Müller (Eds.), *Advances in neural information processing systems*, 12 (pp. 400–406). Cambridge, MA: MIT Press.
- Beniaguev, D., Segev, I., & London, M. (2021). Single cortical neurons as deep artificial neural networks. *Neuron*, 109, 2727–2739. 10.1016/j.neuron.2021.07.002, PubMed: 34380016
- Bi, G.-q., & Poo, M.-m. (1998). Synaptic modifications in cultured hippocampal neurons: Dependence on spike timing, synaptic strength, and postsynaptic cell type. *Journal of Neuroscience*, 18(24), 10464–10472. 10.1523/JNEUROSCI.18-24-10464.1998, PubMed: 9852584
- Bialek, W., Rieke, F., Van Steveninck, R. D. R., & Warland, D. (1991). Reading a neural code. *Science*, 252(5014), 1854–1857. 10.1126/science.2063199, PubMed: 2063199
- Brunner, J., & Szabados, J. (2016). Analogue modulation of back-propagating action potentials enables dendritic hybrid signalling. *Nature Communications*, 7, 13033. 10.1038/ncomms13033
- Carandini, M., & Heeger, D. J. (2012). Normalization as a canonical neural computation. *Nature Reviews Neuroscience*, 13(1), 51–62. 10.1038/nrn3136
- Chen, S. S., & Gopinath, R. A. (2001). Gaussianization. In T. G. Dietterich, S. Becker, & Z. Ghahramani (Eds.), *Advances in neural information processing systems*, 14 (pp. 423–429). Cambridge, MA: MIT Press.
- Child, R. (2020). *Very deep VAES generalize autoregressive models and can outperform them on images*. arXiv:2011.10650.
- Chua, K., Calandra, R., McAllister, R., & Levine, S. (2018). In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, & R. Garnett (Eds.), *Advances in neural information processing systems*, 31 (pp. 4754–4765). Red Hook, NY: Curran.

- Chung, J., Kastner, K., Dinh, L., Goel, K., Courville, A. C., & Bengio, Y. (2015). A recurrent latent variable model for sequential data. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, & R. Garnett (Eds.), *Advances in neural information processing systems*, 28 (pp. 2980–2988). Red Hook, NY: Curran.
- Clark, A. (2013). Whatever next? Predictive brains, situated agents, and the future of cognitive science. *Behavioral and Brain Sciences*, 36(3), 181–204. 10.1017/S0140525X12000477
- Comon, P. (1994). Independent component analysis, a new concept? *Signal Processing*, 36(3), 287–314. 10.1016/0165-1684(94)90029-9
- Cornish, R., Caterini, A. L., Deligiannidis, G., & Doucet, A. (2020). Relaxing bijectivity constraints with continuously indexed normalising flows. In *Proceedings of the International Conference on Machine Learning*. New York: ACM.
- Covic, E. N., & Sherman, S. M. (2011). Synaptic properties of connections between the primary and secondary auditory cortices in mice. *Cerebral Cortex*, 21(11), 2425–2441. 10.1093/cercor/bhr029, PubMed: 21385835
- Cremer, C., Li, X., & Duvenaud, D. (2018). Inference suboptimality in variational autoencoders. In *Proceedings of the International Conference on Machine Learning* (pp. 1078–1086). New York: ACM.
- Crick, F. (1989). The recent excitement about neural networks. *Nature*, 337(6203), 129–132. 10.1038/337129a0, PubMed: 2911347
- Dan, Y., Atick, J. J., & Reid, R. C. (1996). Efficient coding of natural scenes in the lateral geniculate nucleus: Experimental test of a computational theory. *Journal of Neuroscience*, 16(10), 3351–3362. 10.1523/JNEUROSCI.16-10-03351.1996, PubMed: 8627371
- Dasgupta, I., Schulz, E., Goodman, N. D., & Gershman, S. J. (2018). Remembrance of inferences past: Amortization in human hypothesis generation. *Cognition*, 178, 67–81. 10.1016/j.cognition.2018.04.017, PubMed: 29793110
- Dayan, P., & Hinton, G. E. (1996). Varieties of Helmholtz machine. *Neural Networks*, 9(8), 1385–1403. 10.1016/S0893-6080(96)00009-3, PubMed: 12662541
- Dayan, P., Hinton, G. E., Neal, R. M., & Zemel, R. S. (1995). The Helmholtz machine. *Neural Computation*, 7(5), 889–904. 10.1162/neco.1995.7.5.889, PubMed: 7584891
- De Pasquale, R., & Sherman, S. M. (2011). Synaptic properties of corticocortical connections between the primary and secondary visual cortical areas in the mouse. *Journal of Neuroscience*, 31(46), 16494–16506. 10.1523/JNEUROSCI.3664-11.2011, PubMed: 22090476
- Deco, G., & Brauer, W. (1995). Higher order statistical decorrelation without information loss. In D. S. Touretzky, M. C. Mozer, & M. E. Hasselmo (Eds.), *Advances in neural information processing systems*, 8 (pp. 247–254). Cambridge, MA: MIT Press.
- Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39, 1–38. 10.1111/j.2517-6161.1977.tb01600.x
- Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition* (pp. 248–255). Piscataway, NJ: IEEE.
- Dinh, L., Krueger, D., & Bengio, Y. (2015). NICE: Non-linear independent components estimation. In *International Conference on Learning Representations*.

- Dinh, L., Sohl-Dickstein, J., & Bengio, S. (2017). Density estimation using real NVP. In *Proceedings of the International Conference on Learning Representations*.
- Dong, D. W., & Atick, J. J. (1995). Temporal decorrelation: A theory of lagged and nonlagged responses in the lateral geniculate nucleus. *Network: Computation in Neural Systems*, 6(2), 159–178. 10.1088/0954-898X_6_2_003
- Douglas, R. J., Martin, K. A., & Whitteridge, D. (1989). A canonical microcircuit for neocortex. *Neural Computation*, 1(4), 480–488. 10.1162/neco.1989.1.4.480
- Doya, K., Ishii, S., Pouget, A., & Rao, R. P. (2007). *Bayesian brain: Probabilistic approaches to neural coding*. Cambridge, MA: MIT Press.
- Ebert, F., Finn, C., Lee, A. X., & Levine, S. (2017). Self-supervised visual planning with temporal skip connections. In *Proceedings of the Conference on Robot Learning*.
- Egner, T., Monti, J. M., & Summerfield, C. (2010). Expectation and surprise determine neural population responses in the ventral visual stream. *Journal of Neuroscience*, 30(49), 16601–16608. 10.1523/JNEUROSCI.2770-10.2010, PubMed: 21147999
- Eliades, S. J., & Wang, X. (2008). Neural substrates of vocalization feedback monitoring in primate auditory cortex. *Nature*, 453(7198), 1102. 10.1038/nature06910, PubMed: 18454135
- Feldman, H., & Friston, K. (2010). Attention, uncertainty, and free-energy. *Frontiers in Human Neuroscience*, 4. 10.3389/fnhum.2010.00215, PubMed: 21160551
- Fraccaro, M., Sønderby, S. K., Paquet, U., & Winther, O. (2016). Sequential neural models with stochastic layers. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, & R. Garnett (Eds.), *Advances in neural information processing systems*, 29 (pp. 2199–2207). Red Hook, NY: Curran.
- Frey, B. J., Hinton, G. E., & Dayan, P. (1996). Does the wake-sleep algorithm produce good density estimators? In M. Mozer, M. Jordan, & T. Petsche (Eds.), *Advances in neural information processing systems*, 9 (pp. 661–667). Cambridge, MA: MIT Press.
- Friston, K. (2005). A theory of cortical responses. *Philosophical Transactions of the Royal Society of London B: Biological Sciences*, 360(1456), 815–836. 10.1098/rstb.2005.1622
- Friston, K. (2008a). Hierarchical models in the brain. *PLOS Computational Biology*, 4(11), e1000211.
- Friston, K. (2008b). Variational filtering. *NeuroImage*, 41(3), 747–766. 10.1016/j.neuroimage.2008.03.017
- Friston, K. (2009). The free-energy principle: A rough guide to the brain? *Trends in Cognitive Sciences*, 13(7), 293–301. 10.1016/j.tics.2009.04.005, PubMed: 19559644
- Friston, K. (2018). Does predictive coding have a future? *Nature Neuroscience*, 21(8), 1019. 10.1038/s41593-018-0200-7, PubMed: 30038278
- Friston, K., Mattout, J., Trujillo-Barreto, N., Ashburner, J., & Penny, W. (2007). Variational free energy and the Laplace approximation. *NeuroImage*, 34(1), 220–234. 10.1016/j.neuroimage.2006.08.035, PubMed: 17055746
- Gershman, S. (2019). *What does the free energy principle tell us about the brain?* arXiv:1901.07945.
- Gershman, S., & Goodman, N. (2014). Amortized inference in probabilistic reasoning. In *Proceedings of the Cognitive Science Society*. Cognitive Science Society.
- Gidon, A., Zolnik, T. A., Fidzinski, P., Bolduan, F., Papoutsis, A., Poirazi, P., . . . Larkum, M. E. (2020). Dendritic action potentials and computation in human layer 2/3 cortical neurons. *Science*, 367(6473), 83–87. 10.1126/science.aax6239, PubMed: 31896716

- Gilbert, C. D., & Sigman, M. (2007). Brain states: Top-down influences in sensory processing. *Neuron*, 54(5), 677–696. 10.1016/j.neuron.2007.05.019, PubMed: 17553419
- Gillon, C. J., Pina, J. E., Lecoq, J. A., Ahmed, R., Billeh, Y., Caldejon, S., . . . Zylberberg, J. (2021). *Learning from unexpected events in the neocortical microcircuit*. bioRxiv.
- Girard, P., & Bullier, J. (1989). Visual activity in area V2 during reversible inactivation of area 17 in the macaque monkey. *Journal of Neurophysiology*, 62(6), 1287–1302. 10.1152/jn.1989.62.6.1287, PubMed: 2600626
- Girard, P., Salin, P., & Bullier, J. (1991). Visual activity in areas V3a and V3 during reversible inactivation of area V1 in the macaque monkey. *Journal of Neurophysiology*, 66(5), 1493–1503. 10.1152/jn.1991.66.5.1493, PubMed: 1765790
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. Cambridge, MA: MIT Press.
- Graham, D. J., Chandler, D. M., & Field, D. J. (2006). Can the theory of “whitening” explain the center-surround properties of retinal ganglion cell receptive fields? *Vision Research*, 46(18), 2901–2913. 10.1016/j.visres.2006.03.008, PubMed: 16782164
- Graves, A. (2013). *Generating sequences with recurrent neural networks*. arXiv:1308.0850.
- Gregor, K., Danihelka, I., Mnih, A., Blundell, C., & Wierstra, D. (2014). Deep autoregressive networks. In *Proceedings of the International Conference on Machine Learning* (pp. 1242–1250). New York: ACM.
- Gresele, L., Fissore, G., Javaloy, A., Schölkopf, B., & Hyvarinen, A. (2020). Relative gradient optimization of the Jacobian term in unsupervised deep learning. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, & H. Lin (Eds.), *Advances in neural information processing systems*, 33. Red Hook, NY: Curran.
- Grossberg, S. (1987). Competitive learning: From interactive activation to adaptive resonance. *Cognitive Science*, 11(1), 23–63. 10.1111/j.1551-6708.1987.tb00862.x
- Guerguiev, J., Lillicrap, T. P., & Richards, B. A. (2016). *Biologically feasible deep learning with segregated dendrites*. arXiv:1610.00161.
- Gulrajani, I., Kumar, K., Ahmed, F., Taiga, A. A., Visin, F., Vazquez, D., & Courville, A. (2017). Pixelvae: A latent variable model for natural images. In *International Conference on Learning Representations*.
- Ha, D., & Schmidhuber, J. (2018). Recurrent world models facilitate policy evolution. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, & R. Garnett (Eds.), *Advances in neural information processing systems*, 31 (pp. 2450–2462). Red Hook, NY: Curran.
- Hafner, D., Lillicrap, T., Fischer, I., Villegas, R., Ha, D., Lee, H., & Davidson, J. (2019). Learning latent dynamics for planning from pixels. In *International Conference on Machine Learning* (pp. 2555–2565). New York: ACM.
- Harrison, C. (1952). Experiments with linear prediction in television. *Bell System Technical Journal*, 31(4), 764–783. 10.1002/j.1538-7305.1952.tb01405.x
- Hawkins, J., & Blakeslee, S. (2004). *On intelligence: How a new understanding of the brain will lead to the creation of truly intelligent machines*. New York: Macmillan.
- Hebb, D. O. (1949). *The organization of behavior: A neuropsychological theory*. New York: Wiley.
- Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., . . . Lerchner, A. (2017). beta-VAE: Learning basic visual concepts with a constrained variational framework. In *Proceedings of the International Conference on Learning Representations*.

- Hinton, G. E. (2007). How to do backpropagation in a brain. In *NeurIPS Deep Learning Workshop*.
- Hinton, G. E., & Van Camp, D. (1993). Keeping the neural networks simple by minimizing the description length of the weights. In *Proceedings of the Sixth Annual Conference on Computational Learning Theory* (pp. 5–13). New York: ACM.
- Hjelm, D., Salakhutdinov, R. R., Cho, K., Jojic, N., Calhoun, V., & Chung, J. (2016). Iterative refinement of the approximate posterior for directed belief networks. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, & R. Garnett (Eds.), *Advances in neural information processing systems*, 29 (pp. 4691–4699). Red Hook, NY: Curran.
- Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. In *Proceedings of the National Academy of Sciences*, 79(8), 2554–2558. 10.1073/pnas.79.8.2554
- Hosoya, T., Baccus, S. A., & Meister, M. (2005). Dynamic predictive coding by the retina. *Nature*, 436(7047), 71. 10.1038/nature03689, PubMed: 16001064
- Huang, C.-W., Touati, A., Dinh, L., Drozdal, M., Havaei, M., Charlin, L., & Courville, A. (2017). *Learnable explicit density for continuous latent space and variational inference*. arXiv:1710.02248.
- Huang, Y., & Rao, R. P. (2011). Predictive coding. *Wiley Interdisciplinary Reviews: Cognitive Science*, 2(5), 580–593. 10.1002/wcs.142, PubMed: 26302308
- Hyvainen, A., & Oja, E. (2000). Independent component analysis: Algorithms and applications. *Neural Networks*, 13(4–5), 411–430.
- Isaacson, J. S., & Scanziani, M. (2011). How inhibition shapes cortical activity. *Neuron*, 72(2), 231–243. 10.1016/j.neuron.2011.09.027, PubMed: 22017986
- Jehee, J. F., & Ballard, D. H. (2009). Predictive feedback can account for biphasic responses in the lateral geniculate nucleus. *PLOS Comput. Biol.*, 5(5), e1000373. 10.1371/journal.pcbi.1000373, PubMed: 19412529
- Johanning, F. W., Theis, A.-K., Pannasch, U., Ruckl, M., Rudiger, S., & Schmitz, D. (2015). Ryanodine receptor activation induces long-term plasticity of spine calcium dynamics. *PLOS Biology*, 13(6), e1002181. 10.1371/journal.pbio.1002181, PubMed: 26098891
- Jones, I. S., & Kording, K. P. (2020). *Can single neurons solve MNIST? The computational power of biological dendritic trees*. arXiv:2009.01269.
- Jordan, M. I., Ghahramani, Z., Jaakkola, T. S., & Saul, L. K. (1998). An introduction to variational methods for graphical models. *NATO ASI Series D Behavioural and Social Sciences*, 89, 105–162.
- Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82(1), 35–45. 10.1115/1.3662552
- Kanai, R., Komura, Y., Shipp, S., & Friston, K. (2015). Cerebral hierarchies: Predictive processing, precision and the pulvinar. *Phil. Trans. R. Soc. B*, 370(1668), 20140169. 10.1098/rstb.2014.0169
- Keller, G. B., Bonhoeffer, T., & Hübener, M. (2012). Sensorimotor mismatch signals in primary visual cortex of the behaving mouse. *Neuron*, 74(5), 809–815. 10.1016/j.neuron.2012.03.040, PubMed: 22681686
- Keller, G. B., & Mrsic-Flogel, T. D. (2018). Predictive processing: A canonical cortical computation. *Neuron*, 100(2), 424–435. 10.1016/j.neuron.2018.10.003, PubMed: 30359606

- Kessy, A., Lewin, A., & Strimmer, K. (2018). Optimal whitening and decorrelation. *American Statistician*, 72(4), 309–314. 10.1080/00031305.2016.1277159
- Khemakhem, I., Kingma, D., Monti, R., & Hyvarinen, A. (2020). Variational autoencoders and nonlinear ICA: A unifying framework. In *Proceedings of the International Conference on Artificial Intelligence and Statistics* (pp. 2207–2217).
- Kim, Y., Wiseman, S., Miller, A. C., Sontag, D., & Rush, A. M. (2018). Semi-amortized variational autoencoders. In *Proceedings of the International Conference on Machine Learning*. New York: ACM.
- King, P. D., Zylberberg, J., & DeWeese, M. R. (2013). Inhibitory interneurons decorrelate excitatory cells to drive sparse code formation in a spiking model of V1. *Journal of Neuroscience*, 33, 5475–5485. 10.1523/JNEUROSCI.4188-12.2013, PubMed: 23536063
- Kingma, D. P., Salimans, T., Jozefowicz, R., Chen, X., Sutskever, I., & Welling, M. (2016). Improved variational inference with inverse autoregressive flow. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, & R. Garnett (Eds.), *Advances in neural information processing systems*, 29 (pp. 4743–4751). Red Hook, NY: Curran.
- Kingma, D. P., & Welling, M. (2014). Stochastic gradient VB and the variational autoencoder. In *Proceedings of the International Conference on Learning Representations*.
- Körding, K. P., & König, P. (2001). Supervised and unsupervised learning with two sites of synaptic integration. *Journal of Computational Neuroscience*, 11(3), 207–215.
- Krishnan, R. G., Liang, D., & Hoffman, M. (2018). On the challenges of learning with inference networks on sparse, high-dimensional data. In *Proceedings of the International Conference on Artificial Intelligence and Statistics* (pp. 143–151).
- Kumar, M., Babaeizadeh, M., Erhan, D., Finn, C., Levine, S., Dinh, L., & Kingma, D. P. (2020). Videoflow: A flow-based generative model for video. In *Proceedings of the International Conference on Learning Representations*.
- Laparra, V., Camps-Valls, G., & Malo, J. (2011). Iterative gaussianization: From ICA to random rotations. *IEEE Transactions on Neural Networks*, 22(4), 537–549. 10.1109/TNN.2011.2106511, PubMed: 21349790
- Laughlin, S. (1981). A simple coding procedure enhances a neuron's information capacity. *Zeitschrift für Naturforschung c*, 36(9–10), 910–912.
- LeCun, Y., Bengio, Y., & Hinton, G. E. (2015). Deep learning. *Nature*, 521(7553), 436–444. 10.1038/nature14539, PubMed: 26017442
- Lee, D.-H., Zhang, S., Fischer, A., & Bengio, Y. (2015). Difference target propagation. In *Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases* (pp. 498–515). Berlin: Springer.
- Lillicrap, T. P., Cownden, D., Tweed, D. B., & Akerman, C. J. (2016). Random synaptic feedback weights support error backpropagation for deep learning. *Nature Communications*, 7, 13276. 10.1038/ncomms13276, PubMed: 27824044
- Lillicrap, T. P., Santoro, A., Marris, L., Akerman, C. J., & Hinton, G. (2020). Backpropagation and the brain. *Nature Reviews Neuroscience*, 21, 335–346. 10.1038/s41583-020-0277-3, PubMed: 32303713
- London, M., & Häusser, M. (2005). Dendritic computation. *Annu. Rev. Neurosci.*, 28, 503–532. 10.1146/annurev.neuro.28.061604.135703, PubMed: 16033324

- Lotter, W., Kreiman, G., & Cox, D. (2017). Deep predictive coding networks for video prediction and unsupervised learning. In *Proceedings of the International Conference on Learning Representations*.
- Lotter, W., Kreiman, G., & Cox, D. (2018). *A neural network trained to predict future video frames mimics critical properties of biological neuronal responses and perception*. arXiv:1805.10734.
- Maaløe, L., Fraccaro, M., Lievin, V., & Winther, O. (2019). BIVA: A very deep hierarchy of latent variables for generative modeling. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, & R. Garnett (Eds.), *Advances in neural information processing systems*, 32 (p. 8882). Red Hook, NY: Curran.
- MacKay, D. M. (1956). The epistemological problem for automata. In C. E. Shannon & J. McCarthy (Eds.), *Automata Studies* (pp. 235–252). Princeton: Princeton University Press.
- Marino, J., Chen, L., He, J., & Mandt, S. (2020). Improving sequential latent variable models with autoregressive flows. In *Proceedings of the Symposium on Advances in Approximate Bayesian Inference* (pp. 1–16).
- Marino, J., Cvitkovic, M., & Yue, Y. (2018). A general method for amortizing variational filtering. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, & R. Garnett (Eds.), *Advances in neural information processing systems*, 31. Red Hook, NY: Curran.
- Marino, J., Yue, Y., & Mandt, S. (2018). Iterative amortized inference. In *International Conference on Machine Learning*. (pp. 3403–3412). New York: ACM.
- Markram, H., Lübke, J., Frotscher, M., & Sakmann, B. (1997). Regulation of synaptic efficacy by coincidence of postsynaptic APs and EPSPs. *Science*, 275(5297), 213–215. 10.1126/science.275.5297.213, PubMed: 8985014
- McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, 5(4), 115–133. 10.1007/BF02478259
- Mel, B. W. (1992). The clusteron: Toward a simple abstraction for a complex neuron. In J. Moody, S. J. Hanson, & R. Lippmann (Eds.), *Advances in neural information processing systems*, 4. San Mateo, CA: Morgan Kaufmann.
- Meyer, H. S., Schwarz, D., Wimmer, V. C., Schmitt, A. C., Kerr, J. N., Sakmann, B., & Helmstaedter, M. (2011). Inhibitory interneurons in a cortical column form hot zones of inhibition in layers 2 and 5a. In *Proceedings of the National Academy of Sciences*, 108(40), 16807–16812. 10.1073/pnas.1113648108
- Meyer, T., & Olson, C. R. (2011). Statistical learning of visual transitions in monkey inferotemporal cortex. In *Proceedings of the National Academy of Sciences*, 108(48), 19401–19406. 10.1073/pnas.1112895108
- Mnih, A., & Gregor, K. (2014). Neural variational inference and learning in belief networks. In *International Conference on Machine Learning*. (pp. 1791–1799). New York: ACM.
- Mountcastle, V., Berman, A., & Davies, P. (1955). Topographic organization and modality representation in first somatic area of cat's cerebral cortex by method of single unit analysis. *Am. J. Physiol.*, 183(464), 10.
- Mumford, D. (1991). On the computational architecture of the neocortex. *Biological Cybernetics*, 65(2), 135–145. 10.1007/BF00202389, PubMed: 1912004

- Mumford, D. (1992). On the computational architecture of the neocortex: II. *Biological Cybernetics*, 66(3), 241–251. 10.1007/BF00198477, PubMed: 1540675
- Murphy, K. P. (2012). *Machine learning: A probabilistic perspective*. Cambridge, MA: MIT Press, 59.
- Murray, S. O., Kersten, D., Olshausen, B. A., Schrater, P., & Woods, D. L. (2002). Shape perception reduces activity in human primary visual cortex. In *Proceedings of the National Academy of Sciences*, 99(23), 15164–15169. 10.1073/pnas.192579399
- Neal, R. M., & Hinton, G. E. (1998). A view of the EM algorithm that justifies incremental, sparse, and other variants. In M. I. Jordan (Ed.), *Learning in graphical models* (pp. 355–368). Berlin: Springer.
- Oliver, B. (1952). Efficient coding. *Bell System Technical Journal*, 31(4), 724–750. 10.1002/j.1538-7305.1952.tb01403.x
- Olshausen, B. A., & Field, D. J. (1996). Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381(6583), 607. 10.1038/381607a0, PubMed: 8637596
- Olshausen, B. A., & Field, D. J. (1997). Sparse coding with an overcomplete basis set: A strategy employed by V1? *Vision Research*, 37(23), 3311–3325. 10.1016/S0042-6989(97)00169-7, PubMed: 9425546
- Ölveczky, B. P., Baccus, S. A., & Meister, M. (2003). Segregation of object and background motion in the retina. *Nature*, 423(6938), 401–408.
- Palmer, S. E., Marre, O., Berry, M. J., & Bialek, W. (2015). Predictive information in a sensory population. In *Proceedings of the National Academy of Sciences*, 112(22), 6908–6913. 10.1073/pnas.1506855112
- Papamakarios, G., Pavlakou, T., & Murray, I. (2017). Masked autoregressive flow for density estimation. In I. Guyon, Y. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in neural information processing systems*, 30 (pp. 2338–2347). Red Hook, NY: Curran.
- Park, Y., Kim, C., & Kim, G. (2019). Variational Laplace autoencoders. In *Proceedings of the International Conference on Machine Learning* (pp. 5032–5041). New York: ACM.
- Parra, L., Deco, G., & Miesbach, S. (1995). Redundancy reduction with information-preserving nonlinear maps. *Network: Computation in Neural Systems*, 6(1), 61–72. 10.1088/0954-898X_6_1_004
- Parras, G. G., Nieto-Diego, J., Carbajal, G. V., Valdés-Baizabal, C., Escera, C., & Malmierca, M. S. (2017). Neurons along the auditory pathway exhibit a hierarchical organization of prediction error. *Nature Communications*, 8(1), 1–17. 10.1038/s41467-017-02038-6, PubMed: 28232747
- Pearl, J. (1986). Fusion, propagation, and structuring in belief networks. *Artificial Intelligence*, 29(3), 241–288. 10.1016/0004-3702(86)90072-X
- Pitkow, X., & Meister, M. (2012). Decorrelation and efficient coding by retinal ganglion cells. *Nature Neuroscience*, 15(4), 628. 10.1038/nn.3064, PubMed: 22406548
- Poirazi, P., Brannon, T., & Mel, B. W. (2003). Pyramidal neuron as two-layer neural network. *Neuron*, 37(6), 989–999. 10.1016/S0896-6273(03)00149-1, PubMed: 12670427
- Polsky, A., Mel, B. W., & Schiller, J. (2004). Computational subunits in thin dendrites of pyramidal cells. *Nature Neuroscience*, 7(6), 621. 10.1038/nn1253, PubMed: 15156147

- Pourahmadi, M. (2011). Covariance estimation: The GLM and regularization perspectives. *Statistical Science*, 26, 369–387. 10.1214/11-STS358
- Prieto, A., Prieto, B., Ortigosa, E. M., Ros, E., Pelayo, F., Ortega, J., & Rojas, I. (2016). Neural networks: An overview of early research, current frameworks and new challenges. *Neurocomputing*, 214, 242–268. 10.1016/j.neucom.2016.06.014
- Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language models are unsupervised multitask learners. Open AI blog. <https://d4mucfpxsywv.cloudfront.net/better-language-models/language-models.pdf>
- Rao, R. P. (1998). Correlates of attention in a model of dynamic visual recognition. In S. Solla, T. Leen, & K. R. Müller (Eds.), *Advances in neural information processing systems*, 11 (pp. 80–86). Cambridge, MA: MIT Press.
- Rao, R. P., & Ballard, D. H. (1999). Predictive coding in the visual cortex: A functional interpretation of some extra-classical receptive-field effects. *Nature Neuroscience*, 2(1). 10.1038/4580, PubMed: 10195184
- Rao, R. P., & Sejnowski, T. J. (2002). Predictive coding, cortical feedback, and spike-timing dependent plasticity. In R. P. N. Rao, B. A. Olshausen, & M. S. Leewicki (Eds.), *Probabilistic models of the brain*. Cambridge, MA: MIT Press.
- Razavi, A., van den Oord, A., & Vinyals, O. (2019). Generating diverse high-fidelity images with VQ-VAE-2. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, & R. Garnett (Eds.), *Advances in neural information processing systems*, 32 (pp. 14866–14876). Red Hook, NY: Curran.
- Rezende, D. J., & Mohamed, S. (2015). Variational inference with normalizing flows. In *Proceedings of the International Conference on Machine Learning* (pp. 1530–1538). New York: ACM.
- Rezende, D. J., Mohamed, S., & Wierstra, D. (2014). Stochastic backpropagation and approximate inference in deep generative models. In *Proceedings of the International Conference on Machine Learning* (pp. 1278–1286). New York: ACM.
- Rezende, D. J., & Viola, F. (2018). *Taming VAEs*. arXiv:1810.00597.
- Rippel, O., & Adams, R. P. (2013). *High-dimensional probability estimation with deep density models*. arXiv:1302.5125.
- Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6), 386. 10.1037/h0042519, PubMed: 13602029
- Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323, 533–536. 10.1038/323533a0
- Schiess, M., Urbanczik, R., & Senn, W. (2016). Somato-dendritic synaptic plasticity and error-backpropagation in active dendrites. *PLoS Computational Biology*, 12(2), e1004638. 10.1371/journal.pcbi.1004638, PubMed: 26841235
- Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural Networks*, 61, 85–117. 10.1016/j.neunet.2014.09.003, PubMed: 25462637
- Shannon, C. E. (1948). A mathematical theory of communication. *Bell System Technical Journal*, 27(3), 379–423. 10.1002/j.1538-7305.1948.tb01338.x
- Sharma, J., Angelucci, A., & Sur, M. (2000). Induction of visual orientation modules in auditory cortex. *Nature*, 404(6780), 841.

- Sherman, S. M., & Guillery, R. (2002). The role of the thalamus in the flow of information to the cortex. *Philosophical Transactions of the Royal Society of London, Series B: Biological Sciences*, 357(1428), 1695–1708. 10.1098/rstb.2002.1161
- Sønderby, C. K., Raiko, T., Maaløe, L., Sønderby, S. K., & Winther, O. (2016). Ladder variational autoencoders. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, & R. Garnett (Eds.), *Advances in neural information processing systems*, 29 (pp. 3738–3746). Red Hook, NY: Curran.
- Spratling, M. W. (2008). Reconciling predictive coding and biased competition models of cortical function. *Frontiers in Computational Neuroscience*, 2, 4. 10.3389/neuro.10.004.2008, PubMed: 18978957
- Srinivasan, M. V., Laughlin, S., & Dubs, A. (1982). Predictive coding: A fresh view of inhibition in the retina. In *Proceedings of the Royal Society of London. Series B. Biological Sciences*, 216(1205), 427–459.
- Stork, D. G. (1989). Is backpropagation biologically plausible. In *Proceedings of the International Joint Conference on Neural Networks* (Vol. 2, pp. 241–246). Piscataway, NJ: IEEE. 10.1109/IJCNN.1989.118705
- Stuart, G. J., & Sakmann, B. (1994). Active propagation of somatic action potentials into neocortical pyramidal cell dendrites. *Nature*, 367(6458), 69. 10.1038/367069a0
- Stuart, G., Spruston, N., Sakmann, B., & Häusser, M. (1997). Action potential initiation and backpropagation in neurons of the mammalian CNS. *Trends in Neurosciences*, 20(3), 125–131. 10.1016/S0166-2236(96)10075-8, PubMed: 9061867
- Summerfield, C., Egner, T., Greene, M., Koechlin, E., Mangels, J., & Hirsch, J. (2006). Predictive codes for forthcoming perception in the frontal cortex. *Science*, 314(5803), 1311–1314. 10.1126/science.1132028, PubMed: 17124325
- Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, & K. Q. Weinberger (Eds.), *Advances in neural information processing systems*, 27 (pp. 3104–3112). Red Hook, NY: Curran.
- Tabak, E. G., & Turner, C. V. (2013). A family of nonparametric density estimation algorithms. *Communications on Pure and Applied Mathematics*, 66(2), 145–164. 10.1002/cpa.21423
- Tipping, M. E., & Bishop, C. M. (1999). Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 61(3), 611–622. 10.1111/1467-9868.00196
- Uria, B., Murray, I., & Larochelle, H. (2014). A deep and tractable density estimator. In *International Conference on Machine Learning* (pp. 467–475). New York: ACM.
- Vahdat, A., & Kautz, J. (2020). NVAE: A deep hierarchical variational autoencoder. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, & H. Lin (Eds.), *Advances in neural information processing systems*, 33. Red Hook, NY: Curran.
- van den Broeke, G. (2016). *What auto-encoders could learn from brains*. Master's thesis, Aalto University.
- van den Oord, A., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., . . . Kavukcuoglu, K. (2016). Wavenet: A generative model for raw audio. In *Proceedings of the 9th ISCA Speech Synthesis Workshop* (pp. 125–125). International Speech Communication Association.

- van den Oord, A., Kalchbrenner, N., & Kavukcuoglu, K. (2016). Pixel recurrent neural networks. In *Proceedings of the International Conference on Machine Learning* (pp. 1747–1756). New York: ACM.
- van den Oord, A., Li, Y., Babuschkin, I., Simonyan, K., Vinyals, O., Kavukcuoglu, K., . . . others (2018). Parallel wavenet: Fast high-fidelity speech synthesis. In *Proceedings of the International Conference on Machine Learning* (pp. 3915–3923). New York: ACM.
- Van Essen, D. C., & Maunsell, J. H. (1983). Hierarchical organization and functional streams in the visual cortex. *Trends in Neurosciences*, 6, 370–375. 10.1016/0166-2236(83)90167-4
- Von Helmholtz, H. (1867). *Handbuch der physiologischen optik* (Vol. 9). Voss.
- Wacongne, C., Labyt, E., van Wassenhove, V., Bekinschtein, T., Naccache, L., & Dehaene, S. (2011). Evidence for a hierarchy of predictions and prediction errors in human cortex. In *Proceedings of the National Academy of Sciences*, 108(51), 20754–20759. 10.1073/pnas.1117807108
- Walsh, K. S., McGovern, D. P., Clark, A., & O’Connell, R. G. (2020). Evaluating the neurophysiological evidence for predictive processing as a model of perception. *Annals of the New York Academy of Sciences*, 1464(1), 242. 10.1111/nyas.14321, PubMed: 32147856
- Whittington, J. C., & Bogacz, R. (2017). An approximation of the error back-propagation algorithm in a predictive coding network with local Hebbian synaptic plasticity. *Neural Computation*, 29, 1229–1262. 10.1162/NECO_a_00949, PubMed: 28333583
- Widrow, B., & Hoff, M. E. (1960). *Adaptive switching circuits* (Tech. Rep.). Stanford, CA: Stanford University, Stanford Electronics Labs.
- Wiegand, T., Sullivan, G. J., Bjontegaard, G., & Luthra, A. (2003). Overview of the H.264/AVC video coding standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 13(7), 560–576. 10.1109/TCSVT.2003.815165
- Wiener, N. (1942). *The interpolation, extrapolation and smoothing of stationary time series. NDRC report*. New York: New York.
- Wiener, N. (1948). *Cybernetics or control and communication in the animal and the machine*. Cambridge, MA: MIT Press.
- Williams, S. R., & Stuart, G. J. (2000). Backpropagation of physiological spike trains in neocortical pyramidal neurons: implications for temporal coding in dendrites. *Journal of Neuroscience*, 20, 8238–8246. 10.1523/JNEUROSCI.20-22-08238.2000, PubMed: 11069929
- Winkler, C. M., Worrall, D., Hoogeboom, E., & Welling, M. (2019). *Learning likelihoods with conditional normalizing flows*. arXiv:1912.00042.
- Xie, X., & Seung, H. S. (2003). Equivalence of backpropagation and contrastive Hebbian learning in a layered network. *Neural Computation*, 15(2), 441–454. 10.1162/089976603762552988, PubMed: 12590814
- Yamins, D. L., Hong, H., Cadieu, C. F., Solomon, E. A., Seibert, D., & DiCarlo, J. J. (2014). Performance-optimized hierarchical models predict neural responses in higher visual cortex. In *Proceedings of the National Academy of Sciences*, 111(23), 8619–8624. 10.1073/pnas.1403112111
- Yildirim, I., Kulkarni, T. D., Freiwald, W. A., & Tenenbaum, J. B. (2015). Efficient and robust analysis-by-synthesis in vision: A computational framework,

- behavioral tests, and modeling neuronal representations. In *Proceedings of the Thirty-Seventh Annual Conference of the Cognitive Science Society* (Vol. 4). Cognitive Science Society.
- Zador, A. M., Claiborne, B. J., & Brown, T. H. (1992). Nonlinear pattern separation in single hippocampal neurons with active dendritic membrane. In J. Moody, S. J. Hanson, & R. Lippmann (Eds.), *Advances in neural information processing systems*, 4 (pp. 51–58). San Mateo, CA: Morgan Kaufmann.
- Zmarz, P., & Keller, G. B. (2016). Mismatch receptive fields in mouse visual cortex. *Neuron*, 92(4), 766–772. 10.1016/j.neuron.2016.09.057, PubMed: 27974161

Received March 14, 2021; accepted August 14, 2021.