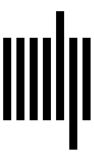
Gradient Expectations

Structure, Origins, and Synthesis of Predictive Neural Networks

By: Keith L. Downing

Citation:

Gradient Expectations: Structure, Origins, and Synthesis of Predictive Neural Networks
By: Keith L. Downing
DOI: 10.7551/mitpress/14723.001.0001
ISBN (electronic): 9780262374675
Publisher: The MIT Press
Published: 2023



The MIT Press

3 Biological Foundations of Prediction

Adaptive behavior governed by predictions begins at the very lowest level of life and continues up through the phylogenetic tree. Gradient detection plays a vital role in survival and arises via pure biochemistry in single-celled organisms and then via neural circuitry in some of the simpler multicellular lifeforms. Many of the same neural principles then enable differentiation and prediction in the brains of higher animals.

This chapter investigates several of these primitive mechanisms, starting with the biochemical and quickly moving up through the neurophysiological to our ultimate goal, the neuroarchitectural. At that level, a diverse collection of neural circuits appears to implement prediction, and in many different ways, but all of which have direct ties to the general concepts introduced in chapter 2. Noticeably absent from this chapter is one dominant predictive region, the neocortex, whose detailed investigation is saved for later chapters on predictive coding and brain evolution, where the building-block modularity of the cortex meshes naturally with both traditional predictive-coding implementations and theories of incremental cognitive emergence.

3.1 Gradient-Following Bacteria

E. coli and other bacteria exploit a simple but effective strategy for moving along nutrient gradients. As shown in figure 3.1, when moving in the direction of increasing resource, the individual maintains a relatively straight line. However, when devoid of a promising gradient, it reverts to a random *tumbling* motion that basically amounts to exploration.

This simple strategy still requires a sophisticated physicochemical process to link chemical gradients to action selection. This begins by using a temporal derivative as a proxy for a spatial derivative: proximal sensory readings of an attractant (A) taken by an agent moving from point s_k (at time t_k) to point s_{k+1} (at time t_{k+1}) allow it to estimate $\frac{\triangle A}{\triangle t}$, which clearly mirrors $\frac{\triangle A}{\triangle s}$ (Dusenbery 1992). Intuitively, if moving in a straight line and detecting an increase in attractant over time, the agent has also recognized an increase across space.

As is common in biological systems, the actual mechanisms do not map cleanly to the computational logic followed by, for example, a gradient-following robot. Rather, the standard engineering variables and calculations exist only implicitly in the natural system. In the case of bacteria, no internal variable explicitly represents $\frac{\triangle A}{\triangle t}$, but the causal relationship between sensors and actuators yields overt gradient-following behavior.

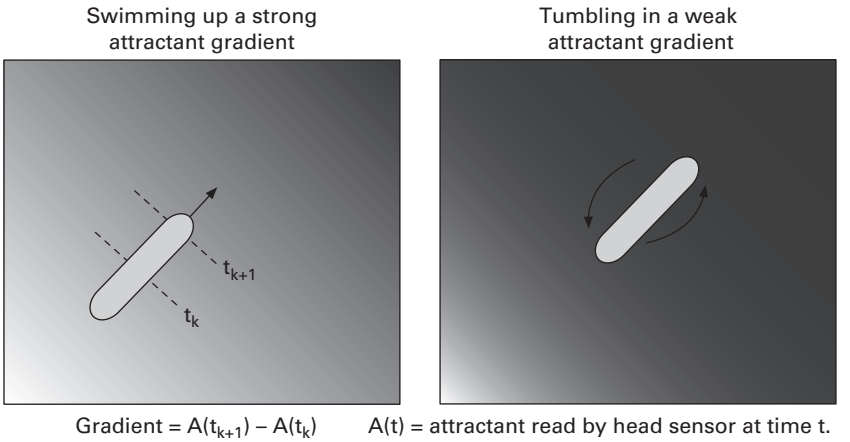


Figure 3.1
Basic swimming and tumbling behaviors employed by gradient-following bacteria (Levit and Stock 2002; Dusenbery 1992). Changes in background shading indicate gradients of chemical attractant.

Figure 3.2 illustrates the basic network of interactions operating within and around the *E. coli*. Both attractants and repellents bind to cell-membrane receptors, with opposite effects on internal kinase activity, which disrupts the coordinated movement of the cell's flagella, thus causing a tumbling motion. In low-kinase situations, the flagella rotate in the same direction and the cell swims in a constant direction. The swimming and tumbling motions constitute an opposing pair. As the diagram shows, attractant inhibits kinase and should thus reduce tumbling and disinhibit straight-line swimming.

However, gradient detection requires additional chemical dynamics. When attractant binds to the receptor, it also promotes the receptor's methylation, yielding it less sensitive to the attractant. This desensitization ensures that continued kinase deactivation will require an increased attractant concentration to bind the more finicky receptor. Thus, keeping kinase and tumbling in check requires an ever-increasing level of attractant.

The relationships among attractant, kinase, and movement appear in the rough sketch of figure 3.3, where (on the left) a relatively stable attractant level loses its inhibitory effects on kinase, and the dominant motion switches from swimming to tumbling. On the right, an ever-increasing attractant concentration holds kinase levels down such that swimming remains dominant.

In summary, these bacteria link ambient chemical concentrations to motion via an opposing pair of flagellar controls mediated by kinase; but tying chemical *gradients* to action requires the additional process of desensitization, which they achieve by receptor methylation. As it turns out, similar mechanisms operate in some of the neural circuits that appear to compute derivatives.

3.2 Neural Motifs for Gradient Calculation

As elaborated by Tripp and Eliasmith (2010), many types of neural circuits can implement gradient detection. Figure 3.4 portrays one of these models, wherein a detector neuron (A) for some quantity (e.g., an attractant) signals another neuron (B), which habituates to A's

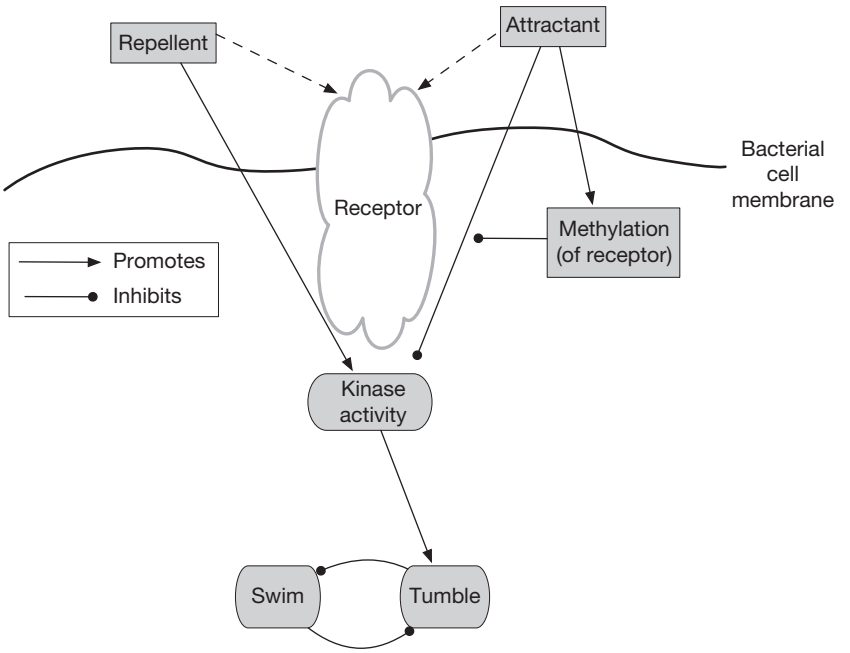


Figure 3.2
Chemical circuits for chemotaxis in *E. coli* bacteria, as described by Levit and Stock (2002).

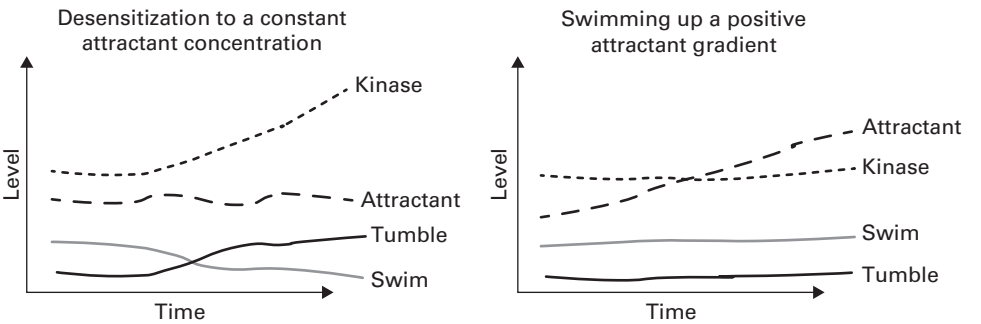


Figure 3.3
Sketch of the general progression of chemical concentrations and activities (swim, tumble) underlying bacterial chemotaxis, based on descriptions in Dusenbery (1992) and Levit and Stock (2002).

input, thus requiring an increase in A's activity for continued firing of B. Detailed analysis of the nematode worm *C. elegans* reveals this model in action (Larsch et al. 2015), albeit with considerable biochemical complexity. The relevant portion of the worm's nervous system, sketched in figure 3.5, shows many connections between a set of sensory neurons and a set of interneurons. Using this network, *C. elegans* can detect chemical gradients that span a concentration range of five orders of magnitude. Two key neurons in this process are AWA and AIA. The former desensitizes to the chemical attractant, while the latter desensitizes to signals from AWA. That combination enables the nematode to swim up very weak (yet nonzero) concentration gradients. In addition, the link from AWC (which is inhibited by the

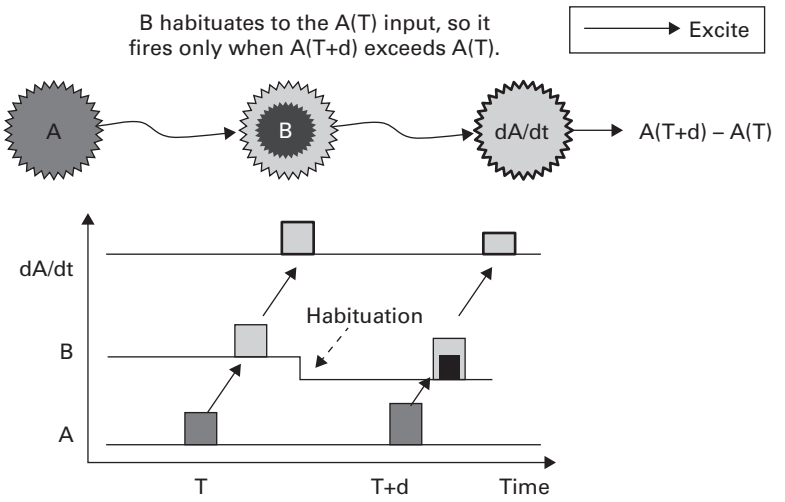


Figure 3.4
A simple neural circuit for detecting a positive derivative of a stimulus. Neuron A responds proportionally to stimulus concentration and excites neuron B, which desensitizes to A's input (indicated by the dark inner circle), thus requiring stronger signals from A in order to fire. The graph displays the temporal progression of this behavior, with the dark inner box on B's plot indicating an elevated firing threshold. The third neuron simply detects B's output and implicitly represents $\frac{\Delta A}{\Delta t}$.

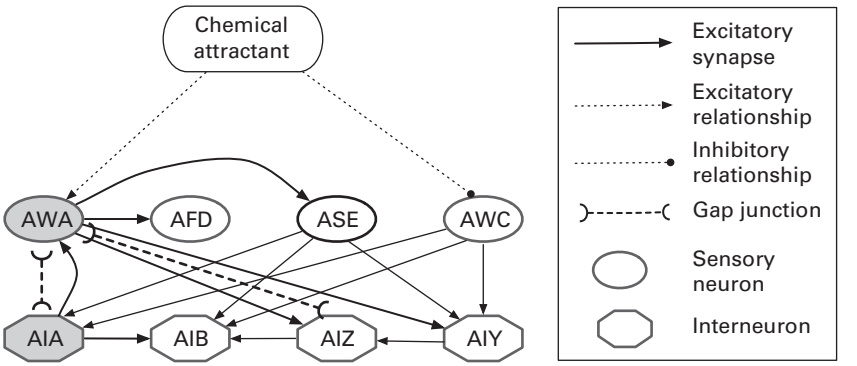
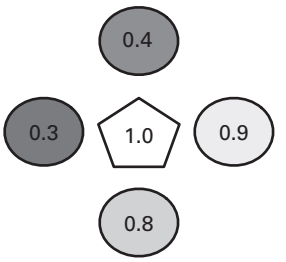


Figure 3.5
A portion of the circuit diagram of the nematode worm *C. elegans*, based on diagrams and descriptions in (Larsch et al. 2015).

chemical attractant) to AIA and other interneurons appears to mediate responses to weakly decreasing gradients, once again via desensitization.

As mentioned above, for simple organisms, one key functional consequence of desensitization is the extended *range* of stimuli that sensory receptors can detect. The mammalian visual system also exploits this relatively simple trick at multiple levels of neural processing to perceive a ten-thousand-fold range of light intensity, despite the fact that standard principles of neural coding admit only approximately a hundred unique neural signals (Dunn, Lankheet, and Rieke 2007). Thus, desensitization allows a sensory system to contextualize inputs such that *contrast* (in time and/or space) carries more salient information than do



$$\text{Prediction} = \frac{0.4 + 0.9 + 0.8 + 0.3}{4} = 0.6$$

$$\text{Reality} - \text{Prediction} = \text{Error}$$
$$1.0 - 0.6 = 0.4$$

$$\text{Contrast} = \frac{\text{Error}}{\text{Prediction}} = \frac{0.4}{0.6} = 0.667$$

Figure 3.6
A central cell (pentagon) and its neighborhood (circles), with numbers denoting luminance in the range 0 (dark) to 1 (bright). Accompanying equations show basic relationships (as further discussed in the text) among the neighborhood average (a prediction of X's value), the difference between X's actual value and that average, and contrast.

absolute stimulus levels; and contrast, a concept with many definitions, clearly represents a gradient in space and/or time.

As portrayed in figure 3.6, contrast ties directly to prediction in common interpretations of neural processing, particularly in the early sensory layers (Sterling and Laughlin 2015). Essentially, the average level of stimulus among neighbors to unit X constitutes a prediction of X's own stimulus level. The difference between X's actual value and this prediction then yields an error term, and scaling that error by the prediction gives a value known as the *Weber contrast*. As discussed more thoroughly in later chapters, this manifests *predictive coding* (Rao and Ballard 1999), which saves considerable resources associated with signal transmission by requiring that only scaled errors (which embody *surprise*) be sent upward in the neural hierarchy. Hence, stimuli that match predictions (and are thus expected), incur very little signaling cost.

This is just one of many incarnations of prediction in the brain. In this case, the relationship between gradient (i.e., contrast) and prediction is reversed in that the former is derived from the latter, and the former becomes the key piece of information sent between neural layers. However, when sent upward, this gradient signal contributes to neighborhood-based predictions in the next level of the neural hierarchy. Contrast exemplifies the *differences that make a difference* touted by Edelman and Tononi (2000) as key elements of neural information processing, and it plays a key role in predictive coding.

Figure 3.7 indicates another mechanism by which neural circuits can detect derivatives (Sterling and Laughlin 2015; Tripp and Eliasmith 2010), in this case via the combination of delay routes and signal inversion. Imagine a simple example involving a series of sensory signals: 3,5,10,7,0,0 received at neuron A from time 1 to time 6. Assume that (a) neurons A and B produce an output signal in direct proportion to their input, but (b) neuron B's signal has an inhibitory effect on its downstream neighbor, and (c) passage through each neuron takes approximately the same nonzero amount of time (d in the diagram). As seen in table 3.1, the network's rightmost output equals the differences between successive inputs, that is, an estimate of the temporal derivative.

Chaining these simple modules together facilitates the computation of higher-order derivatives, as shown in figure 3.8. A few simple modifications to the lower circuit of figure 3.8 produces that of figure 3.9, in which an increase (from 1 to 2) of the weights (w)

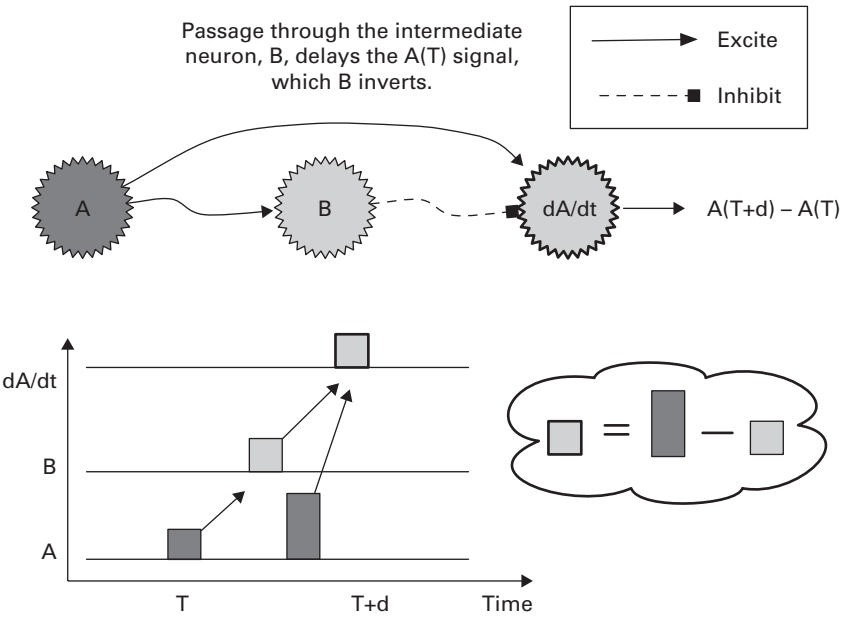


Figure 3.7
Illustration of temporal derivatives computed via a delay pathway and an inverted signal.

Table 3.1
Time series of each neuron's input(s) and rightmost output for the network of figure 3.7 when given input sequence 3,5,10,7,0,0, with the processing time of each neuron (d) being 1.

Time	A	B	$\frac{\partial A}{\partial t}$	Output
1	3	0	[0, 0]	0
2	5	3	[3, 0]	0
3	10	5	[5, −3]	0
4	7	10	[10, −5]	2
5	0	7	[7, −10]	5
6	0	0	[0, −7]	−3

on the leapfrogging connections changes the computed value from a derivative to the previous value plus the derivative: a primitive prediction of the future value ($x + \triangle x$). Thus, by sending downstream its value (weighted by w) and a delayed, inverted version, that neuron's level receives a feedback prediction of its next value.

A few basic simulations, using a set of difference equations that directly model the behavior of the lower circuit of figure 3.8, verify that these networks do indeed produce higher-order derivatives. First, let the generator of input values be a sine curve: $y = sin(kt)$ where t is the integer timestep and k = 0.25. From basic calculus, it follows that $\frac{\partial y}{\partial t} = kcos(kt)$, $\frac{\partial^2 y}{\partial t^2} = -k^2 sin(kt)$, and $\frac{\partial^3 y}{\partial t^3} = -k^3 cos(kt)$. Thus, the k+1st derivative is shifted one quarter phase relative to the kth derivative and has one-fourth the amplitude. The plots of figure 3.10 (left) clearly show the proper shifting and scaling of the outputs produced by the

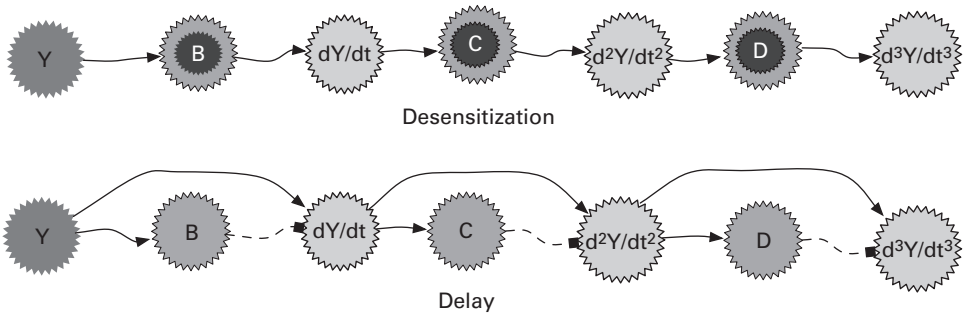


Figure 3.8
Neural circuits for computing three levels of derivatives using desensitization (top) and delayed inhibition (bottom). All connections have weights 1 (excitors) and −1 (inhibitors).

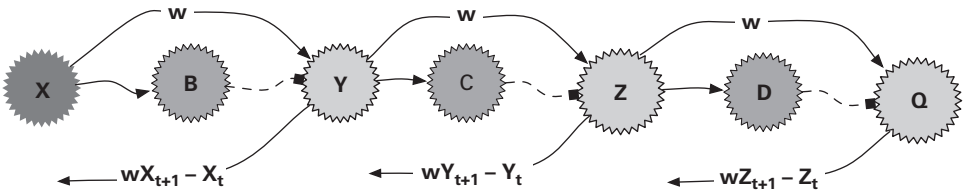


Figure 3.9
A simple neural circuit in which the proper choice of weighting (w = 2) enables the next value of neurons X, Y, and Z to be predicted by layers Y, Z, and Q, respectively. For example, at time 2, Y computes $2X_1 - X_0 = X_1 + (X_1 - X_0) = X_1 + \triangle X_1$. As in figure 3.8 (bottom), neurons B, C, and D act as delayed inverters.

network's three derivative neurons: the network has properly differentiated the input to the third order and could easily continue higher if given more neurons.

As a second example, consider a simple polynomial: $y = \left(\frac{t}{10}\right)^3$. Then, $\frac{\partial y}{\partial t} = \frac{3}{10}\left(\frac{t}{10}\right)^2$, $\frac{\partial^2 y}{\partial t^2} = \frac{6}{100}\left(\frac{t}{10}\right)$, and $\frac{\partial^3 y}{\partial t^3} = \frac{6}{1000}$. As seen in figure 3.10 (right), the delayed-inhibition network accurately computes three orders of derivatives.

In figure 3.10, note that the curves get flatter with the higher-order derivatives, until the third derivative is constant (or nearly so). In other words, at each higher level of the derivative hierarchy, the values change more slowly over time. This is common with polynomials, logarithms, and even exponentials (with fractional exponents such as $y = e^{t/5}$). Thus, for any neural network that implements this derivative hierarchy, the time constants at the higher levels can be higher (i.e., these layers can update more slowly) while still computing accurately.

In fact, the higher-order derivatives normally represent a more abstract, coarser description of the landscape defined by a function: a description that encompasses a larger swath of space and/or time. The first derivative, or slope, describes the incline in the immediate vicinity of a point. The second derivative, or curvature, summarizes the relationship between several inclines, and the third derivative, often called *texture*, denotes the relationship between several curves. For example, a surface with *bumpy* texture consists of many areas of abruptly changing curvature, and the concept of bumpiness normally refers to a region: it is typically not the property of a single or small number of neighboring points.

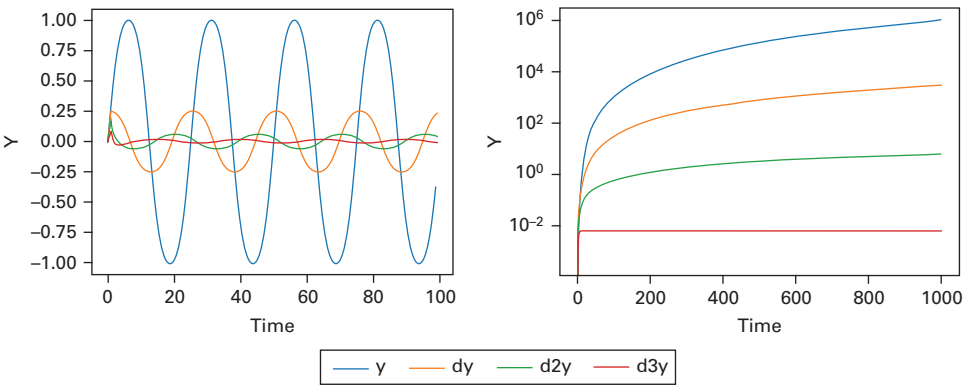


Figure 3.10
Outputs of neuron Y and the three derivative neurons when the delayed-inhibition network (figure 3.8, bottom) processes two different input sequences: $y = sin\left(\frac{t}{4}\right)$ for t = 0, 100 (left), and $y = \left(\frac{t}{10}\right)^3$ for t = 0, 1000 (right).

The plots of figure 3.10 show that reasonably accurate derivatives result from a chain of delayed-inhibition units, but the reverse process should also be possible: the higher levels should be able to drive the lower levels as long as each level receives and integrates inputs from its immediate high-level neighbor. Thus, the highest level should, in essence, predict the values of all lower levels. Additionally, the higher levels should be able to do this while updating at slower timescales. Figure 3.11 illustrates this effect: when the third derivatives corresponding to those from figure 3.10 are fed into the top neuron of figure 3.11 (left) and then propagated downward to neurons (that retain a fraction of their previous value between timesteps), the outputs of each neuron closely mirror those plotted during the inverse process of differentiation from Y upward (in figure 3.10).

When differentiators and integrators are combined into a layered architecture, the integrator at one level can predict the value at its lower neighbor. Figure 3.12 (left) shows a hierarchical composition of these differentiator-integrator pairs, with their key interface being the error unit at each level.

Moving to the right side of figure 3.12, after a brief spin-up period during which the upper levels integrate signals from the lower levels, the network's stable state displays predictive inputs from above that cancel the state values below, yielding small errors. This happens despite the fact that upper levels update much slower than lower levels. In essence, the upper levels project stable long-term expectations (e.g., averages) that give (imperfect but) reasonable estimates of lower-level states, which display more dramatic flux than their above neighbors.

This extremely simple model is intended as no more than a basic abstraction of what a hierarchical brain can do: integrate upward-flowing derivative signals at one level and then use those aggregates as coarse predictions of future states below, while simultaneously providing derivatives of one's own state to even higher levels, running at still slower timescales. The net result is a predictive hierarchy, a more detailed biological version of which appears in the neocortex, as described in chapters 5 and 6.

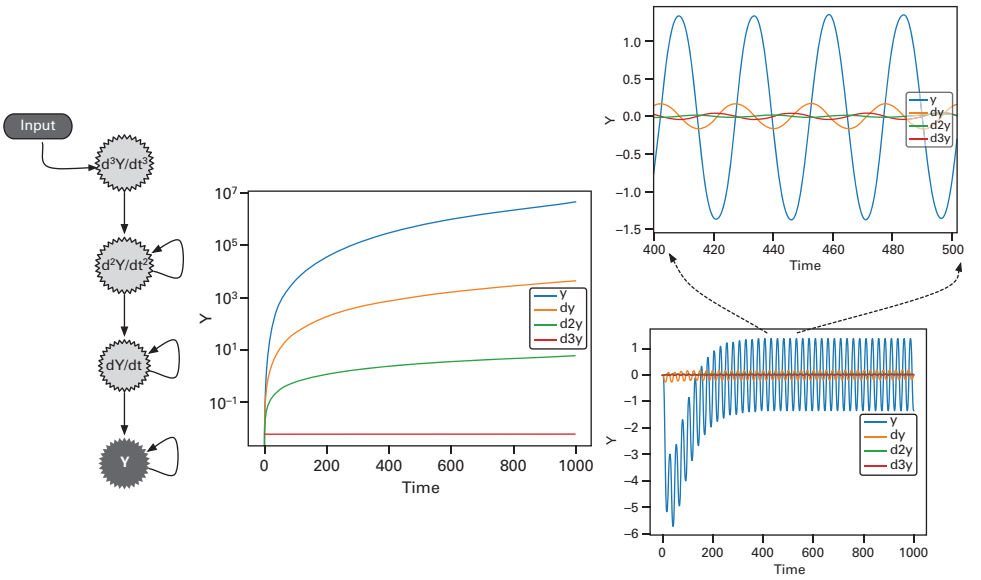


Figure 3.11
(Left) A simple integrator network where weighted output of each upstream neuron combines with a fraction of its downstream neighbor's previous value. Time series (t = 0 to 1000) of each neuron's value when continuously supplied with the following inputs at the top, third-derivative neuron: (Middle) $\frac{\partial^3 y}{\partial t^3} = \frac{6}{1000}$, and (Right) $\frac{\partial^3 y}{\partial t^3} = -\left(\frac{1}{4}\right)^3 cos\left(\frac{t}{4}\right)$. In both runs, the ratios of time constants from Y upward are 1:2:3:4.

3.3 Birth of a PID Controller

As discussed earlier, the combination of derivatives and integrals (sums, averages) directly supports both prediction and control, as encapsulated in the classic equation for a PID controller (repeated from chapter 2):

$$u_t = k_p e_t + k_d \frac{\triangle e_t}{\triangle t} + k_i \sum_{j=0}^{t} e_j \tag{3.1}$$

In short, the error term (difference between goal and current, or predicted, state) undergoes three basic operations (scaling, differentiation, and integration) whose results then combine to produce the control output, which, as detailed earlier, corresponds to a prediction of a future state or error. A simple neural network to perform this calculation appears in figure 3.13.

Nervous systems are replete with all of the primitives that constitute PID and other controllers. To compute error, at least as a first approximation, a network merely needs a comparator that subtracts one input (e.g., the prediction) from another (e.g., the target). Hence, if predictive synapses inhibit the comparator while target synapses excite it, the combination embodies a comparison that yields predictive error. To integrate signals over a longer time frame than that of a single depolarizing or spiking event, an individual neuron merely needs physicochemical properties that dictate a larger time constant (than that of a spike). These *slower* neurons abound in all nervous systems. Another common neural

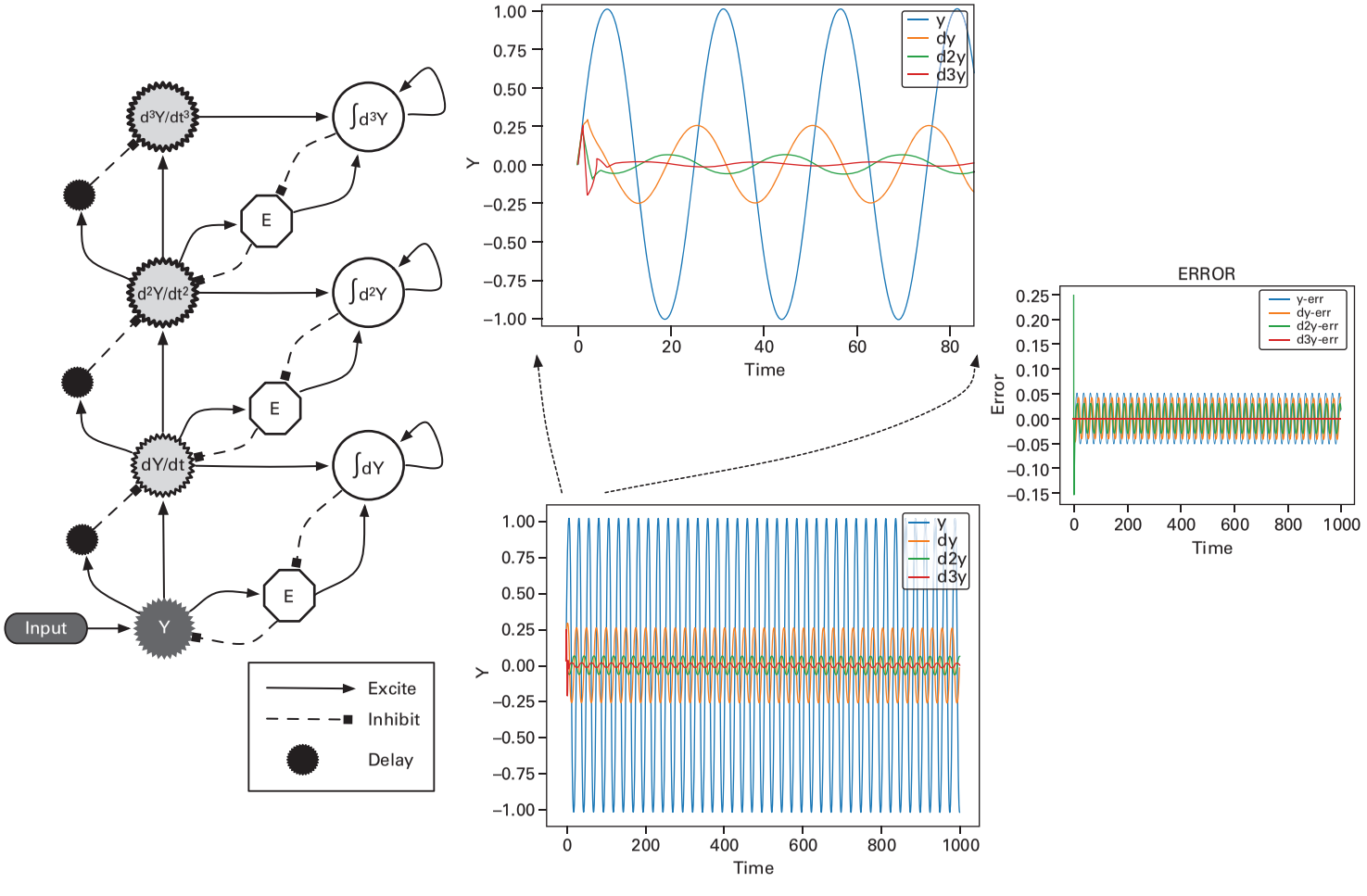


Figure 3.12
(Left) A simple network that combines delayed-signal derivatives and integrators (of derivatives) to perform predictive coding; the E nodes compute prediction error. (Right) Results of the network when fed a time series of values: $y = sin\left(\frac{t}{4}\right)$ for t = 0, 1000. The ratios of time constants from Y upward are 1:4:8:16, indicating that the upper-layer integrator $\int \frac{\partial^3 y}{\partial t^3}$ updates at $\frac{1}{16}$ the speed at which the bottom layer receives inputs. The upper plot (of Y and its derivatives) is a magnified portion of the first 80 timesteps of the lower plot, while the rightmost plot is of the error units over the full 1,000 timesteps.

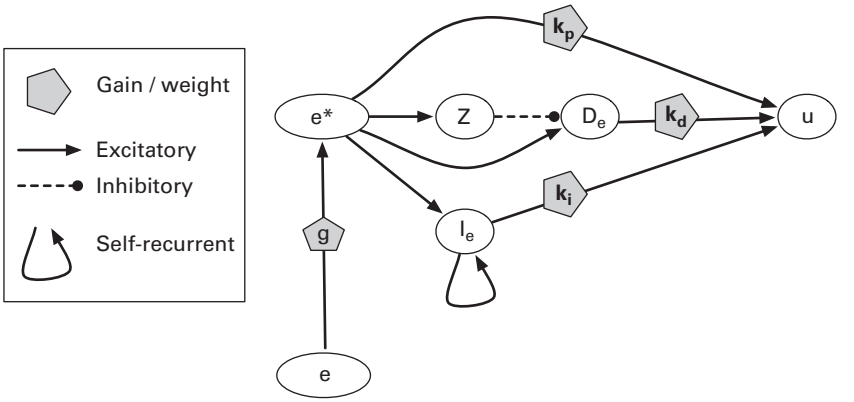


Figure 3.13
A neural network implementation of a PID controller. The original error (e) is scaled by a gain/weight (g) during transfer to gateway error neuron, e*. Each of the constants from equation 3.1 appear as weights on connections to the output neuron, u. The self-recurrent loop of neuron I_e performs basic integration, since I_e's output from timestep t becomes part of the input at step t+1. The delayed inhibition in the e*-Z-D_e chain computes a derivative.

mechanism for caching activation histories is recurrence, wherein neuron A feeds into neuron B, which then feeds back to A. The time lag involved in this double transfer ensures that A receives information colored by its past activity, which it combines with signals from other sources that are more strongly representative of the present. As for the constant terms (k_*) in figure 3.13, these are captured by the total number, location (proximal or distal), and strength of synapses linking one unit, such as a comparator, to another, for example, an integrator.

Perhaps most interesting are the delay lines that facilitate gradient calculations. As discussed below, brains are a mixture of excitatory and inhibitory neurons (at an overall ratio of about 5:1, with excitation in the majority (Kandel, Schwartz, and Jessell 2000)), and most local neural circuits also contain an assortment of both types. When axons from excitors of area A grow into area B, they may target B's excitors, inhibitors, or both. And the inhibitors tend to synapse on the excitors (along with other inhibitors). Collectively, these connections support a circuit similar to that involving e*, Z, and D_e in figure 3.13: the direct connections from A to B's excitors mirrors the e*-D_e link, while the inhibitors of B resemble Z in that they both delay and invert the signal from A's to B's excitors. The inhibitors delay transmission simply by being one extra link in the synaptic chain from A's excitors to B's. Of course, stringing several such circuits together (A to B to C) can realize higher-order derivatives as well.

3.3.1 Adaptive Control in the Cerebellum

One area of the brain often characterized as a controller is the cerebellum, which has a well-established role in the learning and control of complex motions (Kandel, Schwartz, and Jessell 2000; Bear, Conners, and Paradiso 2001), as well as cognitive skills such as attention and speech production. In general, it seems to have a lot to do with timing aspects of various skills; and because of the inherent delays in nervous systems, proper timing requires prediction.

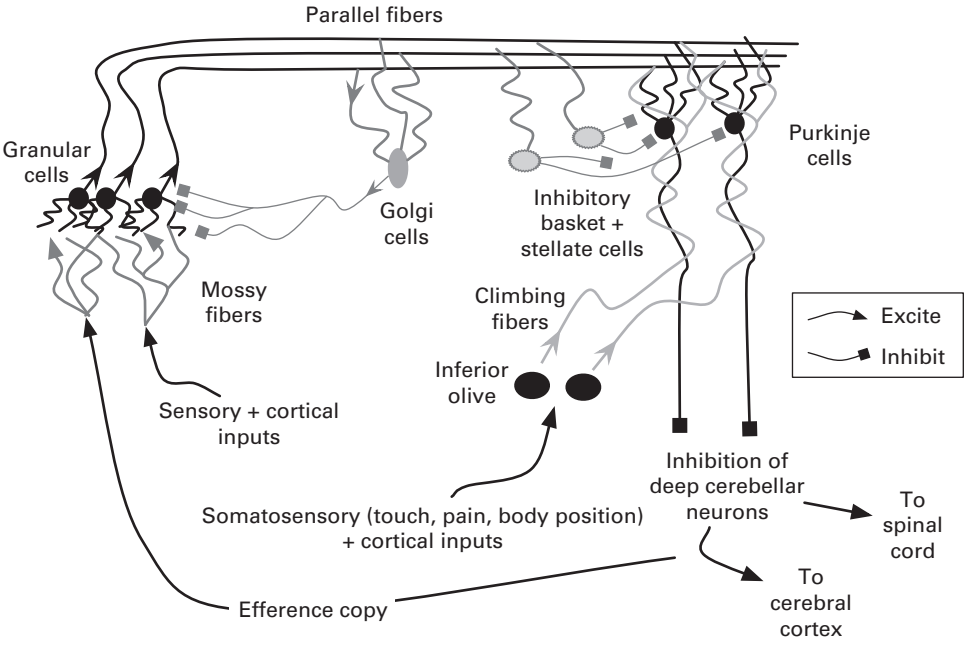


Figure 3.14
Basic anatomy of the mammalian cerebellum, based on images and diagrams in Kandel, Schwartz, and Jessell (2000); Bear, Conners, and Paradiso (2001); and Rolls and Treves (1998).

As shown in figure 3.14, the cerebellar input layer, consisting of granular cells, receives a variety of peripheral sensory and cortical signals via mossy fibers stemming from the spinal cord and brainstem, and realizing a mixture of delays. As the most abundant neuron type in the mammalian brain—the human cerebellum contains approximately 10^{11} (Kandel, Schwartz, and Jessell 2000)—these granular cells appear to manifest expansion coding of sensory and *corollary-discharge* signals, that is, efference copies of motor commands. The tendency of granulars to laterally inhibit one another (via nearby golgi cells) characterizes them as sparse-coding context detectors (Rolls and Treves 1998). Since delay times vary along the mossy fibers, each context has both temporal and spatial extent.

One parallel fiber (PF) emanates from each granular cell and synapses onto the dendrites of many Purkinje cells (PCs), each of which may receive input from 10^5 to 10^6 parallel fibers (Kandel, Schwartz, and Jessell 2000). The PFs also synapse onto basket and stellate cells, both of which inhibit nearby PCs. Hence, the granulars can have both a direct excitatory effect on Purkinje cells and a delayed inhibitory influence. As described above, this allows PCs to detect temporal gradients of the contexts represented by the granulars.

Since the Purkinje outputs are the cerebellum's ultimate contribution to motor and cognitive control, the plethora of granular inputs to each Purkinje cell appears to represent a complex set of preconditions for the generation of any such output. These antecedent-consequent rules are adaptable, since the PF-PC synapses yield to both long-term potentiation (LTP) and (more prominently) long-term depression (LTD) (Kandel, Schwartz, and Jessell 2000; Rolls and Treves 1998; Porrill and Dean 2016). Adaptation, particularly LTD, of the PF-PC synapses is triggered by inputs from climbing fibers of the inferior olive, which transfer

information such as pain signals from the muscles and joints directly influenced by those fibers' corresponding PCs. Climbing fibers exhibit a simple version of supervised learning (Doya 1999) wherein the olivary pain signal modulates the PC's output (which appears to manifest a combination of tracking and prediction error) to drive LTD of the PF-PC synapses. In this way, any combination of predicted-state, goal-state, and sensory information that produces a particular action (that leads to an immediately undesirable outcome) will have a reduced ability to incite that action in the future.

Although learning via the PF-PC synapses involves both LTP and LTD (Porrill and Dean 2016), the latter has garnered the most attention in neuroscience research. The prominence of LTD may stem from the sheer density of afferent parallel-fiber synapses onto each Purkinje cell, and thus the need to adaptively neutralize as many as possible to form viable context-action rules. This hints of Edelman's (1987) *neural selectionism* (discussed in chapters 6 and 7), in which many synapses form early in life but are then gradually pruned by the LTD induced by the agent's experiences.

It is also worth noting that outputs from PCs have inhibitory effects on their postsynaptic neighbors. Thus, the arduous process of tuning brains to achieve motor and cognitive skills involves figuring out *what to turn off* in a given context. This paints the skeletomuscular system as a collection of overeager motor units that are gradually tamed and coordinated by the cerebellum. Somewhat counterintuitively, the simplest behaviors often require the most complex neural activity patterns. For example, it takes a much more intricate combination of excitatory and (particularly) inhibitory signals to wiggle a single finger (or toe) than to move all five. Hence, the tuning of PC cells to achieve the appropriate inhibitory mix is a critical factor in basic skill learning.

The earliest cybernetic models of the cerebellum date back to the work of David Marr (1969) and James Albus (1971). Known as the *Marr-Albus model*, it has many interpretations, including that of an adaptive controller (Fujita 1982), which employs a control element known as an adaptive filter (aka forward model) to predict future states given current states and actions. As pointed out by Porrill and Dean (2016) in their investigation into the Marr-Albus and other models of cerebellar control, contemporary neuroscientific evidence still gives modelers considerable interpretive freedom. The following brief description exercises a bit of that flexibility while illustrating the use of prediction and control in cerebellar function.

Figure 3.15 portrays a hybrid anatomical-functional model of the cerebellum, with abstractions of the basic topology framing the functional modules. Be aware that any attempts to map controller components to the anatomy of any brain region involve a great degree of speculation.

Starting with the mossy fibers and granular cells, their inputs include many aspects of context: sensory (including proprioceptive) signals, desired / goal states of the brain-body-environment coupling, and corollary discharges (aka efference copies) from motor and premotor regions that indicate the impending action. Since this region appears to integrate information across time, via both the differential delay lines on the mossy fibers and the recurrent loop from granulars to parallel fibers to golgi cells and back to granulars, it seems well-equipped to make predictions of future states based on a recent time window of sensory input plus the efferent motor copy. Thus, this area constitutes a forward model that can be classified as an adaptive filter if several of its parameters admit tuning. And, indeed,

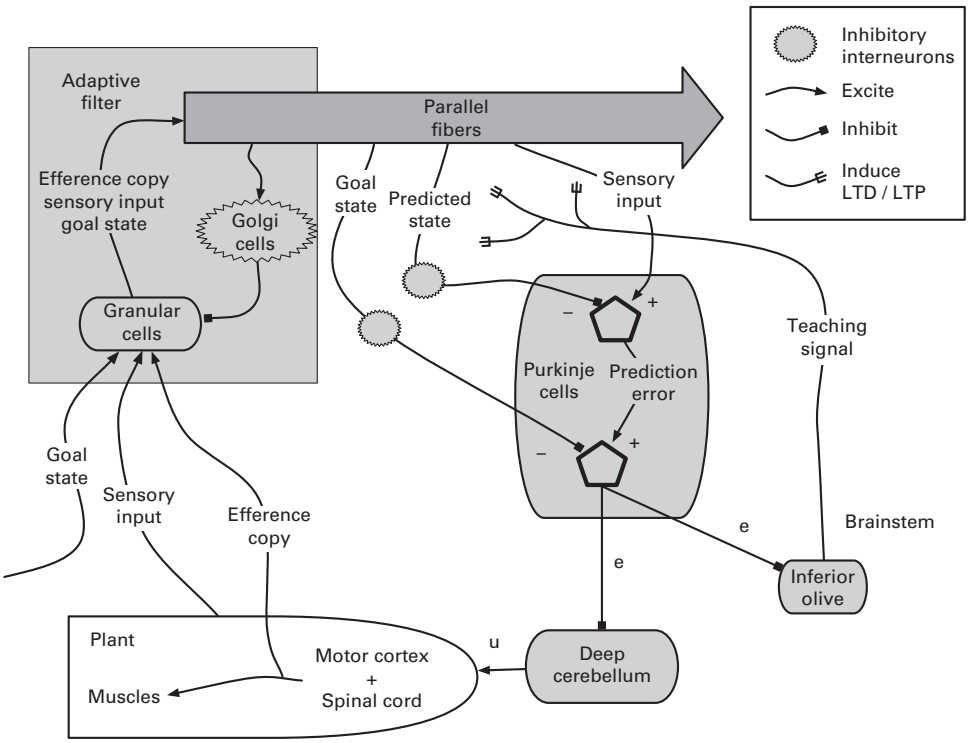


Figure 3.15
Model of cerebellum as an adaptive controller. Controller modules are shaded, while the animal's body, which constitutes the *plant* of cybernetics, is unshaded. Locations of the comparators, adaptive filter, control error (e), and control output (u), are estimates; these may be more widely distributed throughout the cerebellum and brain as a whole.

many synapses in the cerebellum admit LTP and LTD. Since the main evidence of cerebellar synaptic change comes from the PF-PC link, these too might be included in the adaptive forward model. At any rate, by the time signals reach the soma of Purkinje cells, they should encode a predicted state along with the current state and goals.

The Purkinje cells then act as comparators. Since PCs do not feed into one another in series, the vertical string of comparisons shown in figure 3.15 would need to invoke the rich web of PC dendrites, which many researchers propose as the source of complex computations well beyond that of simply transferring all afferent signals to the soma for summation (Hawkins and Ahmad 2016). Taken in succession, for ease of explanation, the first comparison involves the most recent sensory information (constituting a current state) and a prediction of that state (based on an earlier state and the efference motor copy).

For many tasks, such as maintaining stable views of the environment despite head movement, the prediction (e.g., of image movement) derived from the impending action (e.g., head rotation), must be subtracted from the raw image (of drastically shifting surroundings) to yield the actual perception (of a stationary world). Thus, subtraction of a prediction (including the efference copy) from the raw sensory reality produces an accurate picture of the current situation. The Purkinje area's second comparison involves the prediction-adjusted state and the goal, giving the ultimate output error. This is the standard error term

in a feedback controller, the one used to determine the next action. In figure 3.15, this error information emanates to deeper cerebellar areas, where it is converted into a control signal (u) for directing motor activity. Sensory signals from the body along with the efference copy from the motor system then complete the feedback loop via the mossy fibers.

Evidence of the Purkinje cells' role as comparator of predictions, goals, and efference copies to sensory reality extend back in evolutionary history to a diverse collection of primitive (extant) fishes, all of which exhibit parallel fibers and Purkinje cells in cerebellar-like structures. As described by Bell and colleagues (1997), the inhibitory stellate neurons near the PF-PC interface can invert prediction signals such that the net Purkinje output embodies a comparison of reality to expectations. In addition, these proto-cerebellums display climbing fibers that initiate anti-Hebbian modifications to the PF-PC synapses.

3.4 Detectors and Generators

Up to this point, we have viewed prediction in terms of scalar values, their derivatives, and their integrals. This may apply to some simple nervous systems, where the outputs of isolated neurons have significant semantic value, but most advanced neural systems employ population codes: the activity levels of hundreds or thousands of neurons may constitute a salient signal or representation. Patterns in one network region (e.g., layer) can then cause patterns in another, with designations of these regions as higher or lower than one another (though sometimes arbitrary) framing the activity as bottom-up or top-down.

For example, figure 3.16 (left) displays a bottom-up scenario in which a pattern appears at a lower level and then promotes firing of a detector neuron (Z) due to strong synaptic connections between the three constituents and Z. Conversely, Z can function predictively if its activity causes neurons A, D, and E to activate, as shown on the right of the figure. The generative direction also characterizes motricity, wherein higher-level activities lead to the firing of motor neurons and the contraction of particular combinations of muscles.

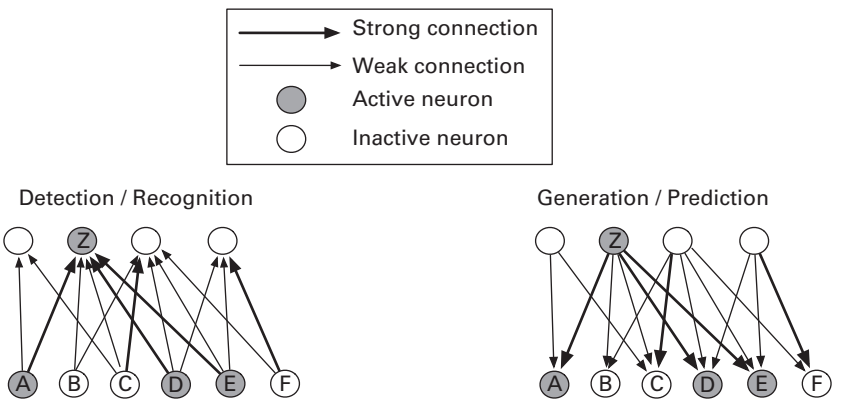


Figure 3.16
Two directions of signal flow in a neural system. (Left) Bottom-up transmission from, for example, sensory levels to higher regions. Neuron Z detects pattern A-D-E. (Right) Top-down signaling from higher to lower levels. Neuron Z predicts pattern A-D-E.

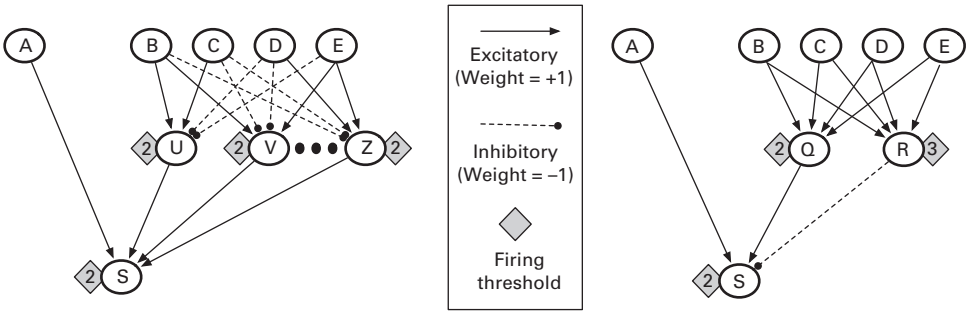


Figure 3.17
Simple neural networks for detecting the concept *A and exactly two of the other four inputs*, where inputs A–E have binary values (0 or 1) and feed into nonlinear neurons that output a 1 if and only if their sum of weighted inputs is equal to or above the associated firing threshold. (Left) A complex network solution that checks each of the six mutually exclusive pairs of B–E. (Right) A simpler network that requires Q to fire (indicating two or more of B–E) and R to remain silent (indicating that less than three of B–E have fired).

As the patterns to detect or predict become more complex, so too do the neural circuits for handling them. One key to complexity is a mixture of excitation and inhibition: too much of one or the other leads to all-or-nothing activity across a neural region, a situation that typically conveys little useful information. Although a single node in an artificial neural network may stimulate some neurons and inhibit others, these two activities normally require different types of neurons in brains. Thus, an excitatory neuron can inhibit another neuron only by first stimulating an inhibitory interneuron. The sample networks described below follow the *artificial convention* to simplify the diagrams, but be aware that inhibitory connections would actually require additional intervening neurons in a biological network.

Multiple layers of neurons provide another complexity enhancement. Consider the circuits of figure 3.17, where unit S acts as the detector for a particular activity pattern among the five sensory units. This computation requires excitatory and inhibitory interneurons, although their cardinality and connection topology can vary. On the left of the figure, the interneurons U–Z each detect one of the six possible combinations of B–E. Notably, this demands many inhibitory connections to ensure that exactly two of the four sensors have fired. However, a simpler circuit (on the right), with only one inhibitory link, also solves the problem. This requires neuron Q to detect *two or more* and neuron R to detect *three or more* such that neuron S fires when A and Q fire, but R does not. Imagine the savings in both interneurons and inhibitory connections, by using a slightly modified version of the rightmost circuit, if the problem were expanded to *A and any three of the other fifty*. A few logical tricks can save a lot of resources when building detectors.

The situation gets more complicated for generators. The circuits in figure 3.18 are designed to activate A and exactly two of B–E whenever unit S fires. Think of this as playing a chord on an instrument such as a clarinet, where A controls the thumb, which needs to remain active to balance the instrument, while two of the four fingers need to change position to open or close particular holes. The difficulty stems from the inversion of the detector problem, which mapped many sensory possibilities to one detector. Now, the mapping goes from one *intention to act* to many motor alternatives, only one of which can activate. This type of circuit typically requires a lot of inhibition along with more precise timing considerations.[1]

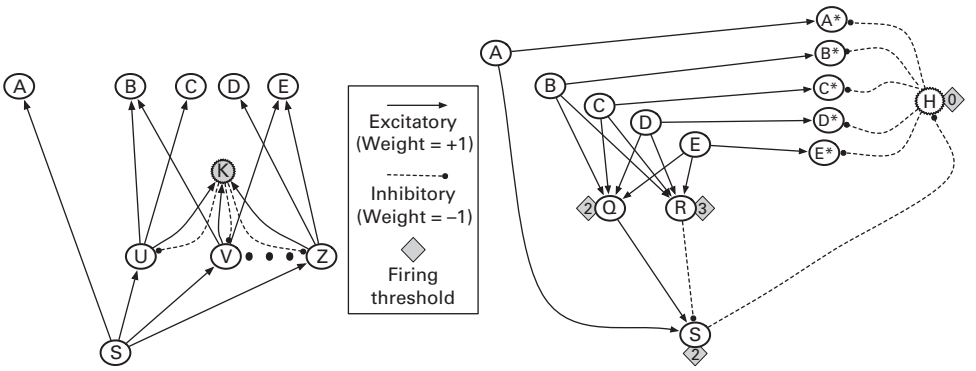


Figure 3.18
The inverse situation of figure 3.17: Each cell A–E represents a motor unit, and when S activates, it should trigger a complex activity in which unit A activates along with exactly two of the B–E units. (Left) Each of the U–Z units codes for one unique pair of the B–E units. The signal from S presumably activates U–Z asynchronously such that exactly one of them (e.g., U) fires first. That triggers the B-C pair while also exciting neuron K, which immediately inhibits all of the U–Z neurons. Thus, timing and widespread inhibition become crucial. (Right) Units A–E now represent premotor neurons that stimulate the corresponding motor units A*–E* (each of which has a firing threshold of 1). As in figure 3.17, unit S functions as a recognizer of the *A and exactly two of B–E* pattern. Premotor units A–E fire randomly, with no motor effects until S fires, thus *unlocking*, via disinhibition across H, the motor units, which require input from their corresponding premotor unit to fire. Inhibitor neuron H requires no input to remain active. Note that this scales linearly to any number of premotor and motor units, with each new pair requiring a small constant number (4) of additional connections.

One approach (figure 3.18, left) involves interneurons U–Z, which again represent each of the mutually exclusive alternatives; but proper behavior now depends on asynchronous firing and fast inhibition of the later-activating alternatives, as explained in the figure caption. This solution scales very poorly (as does the leftmost detector circuit in figure 3.17) for networks with many motor units. It seems unlikely that any one-to-many circuit could perform this operation without many interneurons and considerable inhibition.

Fortunately, the problem can be reformulated as one of detection and disinhibition, as shown on the right of figure 3.18. Now units A–E function as randomly firing premotor neurons, whose *intended* activation pattern runs through units Q, R, and S, with S once again acting as a detector that now *gates* the premotor intention forward to the motor units, A*–E*, in much the same way the basal ganglia appears to gate premotor intentions through disinhibition (Houk, Davis, and Beiser 1995). In this case, the premotor signals are blocked from producing motor activity until the gating condition is satisfied. Thus, instead of working from the default assumption in connectionism that all units are inactive until stimulated by input or an upstream neighbor—an assumption that works well for pattern detectors—a network designed to produce complex actions benefits from assuming the random percolation of intentional units whose downstream signals are normally blocked but occasionally released (disinhibited).

This approach scales well, as illustrated by figure 3.19, where one network can realize different intricate actions, with each requiring only a different pattern detector, not a combinatorial explosion of interneurons and inhibitors. However, this design has at least one major flaw: temporal overlap between valid and invalid patterns. For example, assume that the premotor pattern A-B-C has just fired. This triggers unit S and thus disinhibits *all* of the motor units, but only three of them, A*, B*, and C*, have enough positive stimulation to

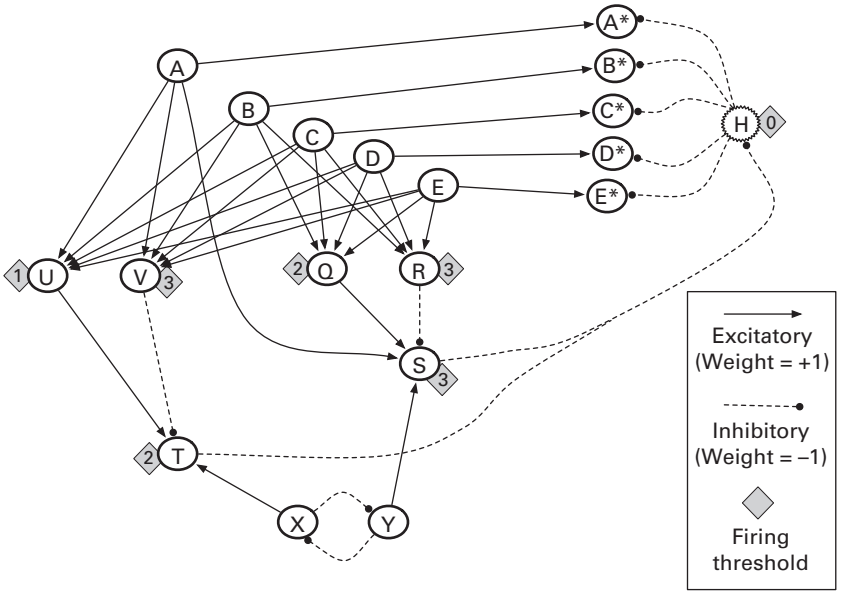


Figure 3.19
A network for evoking one of two mutually exclusive patterns using motor units A*–E*: (1) Activate exactly one or two of the motor units (when unit X fires), and (2) activate unit A* and exactly two of units B*–E* (when unit Y fires). This assumes random firing of the premotor units, A–E, such that when X or Y fires, the detector T or S, respectively, has enough base stimulation to fire if and only if its other afferents (U and V, or Q and R, respectively) have the appropriate settings. Firing of S or T then deactivates the inhibitor unit, H, thus releasing all motor units (which have a firing threshold of 1) from inhibition and allowing any stimulated by their premotor counterpart to fire.

actually fire. So far so good. But what if unit D spontaneously fires just after A-B-C and S? Since disinhibition spans all motor units, D* would also fire, producing an improper motor activity.

To remedy these types of problems, brains have evolved a wide range of oscillatory patterns, from 0.01 to 600 Hz (Buzsaki 2006). Clusters of neurons firing (more or less) synchronously, but without external influence, constitute internal rhythm generators whose periodic signaling governs the firing behavior of their efferents. In figure 3.20, the Z cluster represents an oscillator whose periodic inputs to premotor units A–E is just enough to push them to their firing thresholds in situations where they have also been stimulated by one other pathway (or by spontaneous depolarization). Hence, for units A, B, and C to fire unit S, they must first reach their firing thresholds with the help of Z's input, which only occurs periodically, at the top of each cycle. When this happens, all motor units briefly disinhibit and A*–C* can fire. However, when unit D exhibits spontaneous activity shortly afterward, the Z cycle has already declined, and D cannot fully depolarize and therefore cannot stimulate motor unit D*.

Essentially, the oscillator *sweeps up* all units that have been partially active during its ascending cycle by infusing them with enough extra signal to push them over their thresholds, thus gating a pattern forward (to the detectors and motor units) via a synchronous round of premotor firing. But any premotor units that begin depolarizing after the peak (e.g., unit D in the figure) will either die out, due to the short latency of their spontaneous activity, or have to wait until the next peak (if they can maintain their semi-depolarized state).

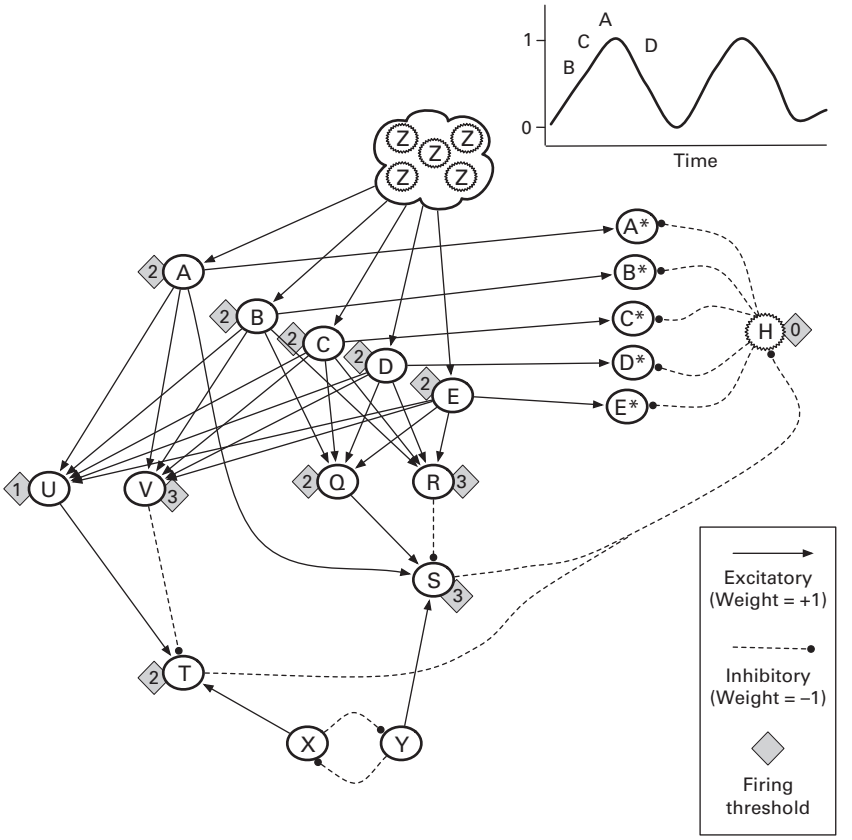


Figure 3.20
A multiple-action network in which premotor units A–E require input from the oscillatory neuron cluster (Z's) along with their own spontaneous activity to reach their firing threshold of 2. The graph in the upper right tracks the synchronous activity of the oscillating cluster, which fires (sending a 1 to the premotor units) only at the peak of each cycle. The letters A–D on the graph indicate the time points at which premotor units A–D are spontaneously active but not spiking (i.e., depolarizing but not yet firing due to a higher threshold than that exhibited by the same neurons in figure 3.19.)

Beyond this simple example, in real brains, oscillators of varying frequencies can sweep up signals exhibiting more or less temporal disparity. A low-frequency rhythm can bundle semi-active neurons over a broad temporal window (assuming that each has a slow time constant), whereas quick oscillators impose stricter time constraints on potential cell ensembles. Thus, given the basic relationships between time and space, a low-frequency cycle can help coordinate the firing of neurons that interact over longer distances but would have a hard time synchronizing otherwise. In short, the brain's cycles both enable and enforce synchronicity such that the neural network can operate in a more discrete fashion, gating in well-established activity patterns with a minimum of noise.

When viewing the brain as a predictive machine, the top-down generative phase of network behavior takes center stage. Since most high-level concepts can be realized by a combinatorial explosion of lower-level patterns, the circuitry for prediction in living organisms probably demands considerable inhibition, disinhibition, and oscillation to ensure that only one of these options gets chosen at any one time, without interference from other

patterns. Thus, at first glance, it appears that the computational demands of generation / prediction exceed those of detection / recognition. However, in theory, the brain must also confront the fact that an infinite number of interpretations exist for any sensory state, so sorting them out may also require its fair share of resources.

3.4.1 The Hippocampus

In mammals, the cerebral hemispheres comprise the neocortex, at the top, and three main subcortical structures: amygdala, basal ganglia, and hippocampus (Kandel, Schwartz, and Jessell 2000). The neocortex is a thin sheet of six-layered cortical columns, which chapter 5 explores extensively as a sophisticated, modular architecture for predictive coding.

Although reptiles, amphibians, and birds lack a neocortex, they have cranial regions analogous to these three subcortical areas (Striedter 2005). In particular, they all have a hippocampus (HC), which is essential for memory formation (Squire and Zola 1996; Andersen et al. 2007). The neocortex is often viewed as the *crowning* (literally) achievement of mammalian brain evolution, the cerebral matter that (somewhat) justifies the moniker *higher intelligences* bestowed on humans, monkeys, and even rats. Although the hippocampus resides below the neocortex, and is more primitive evolutionarily, it seems to occupy the top layer of the *functional* cortical hierarchy; and it exhibits a very sophisticated form of predictive coding.

Comparative brain anatomy (Striedter 2005) clearly reveals that inputs to HC come less from low-level brain regions and more from higher regions as one ascends from amphibians to reptiles to birds and finally to mammals (Striedter 2005), where the entorhinal cortex (EC) serves as an exclusive gateway to HC: almost all signals going into and out of HC go through EC (Rolls and Treves 1998; Andersen et al. 2007). Neural firing patterns in EC have already been through many levels of processing, so they represent reasonably abstract, multimodal (i.e., involving combinations of visual, auditory, haptic, olfactory, and so on) information. When fed into HC from EC, these signals distribute to several different areas, but with different effects.

As shown in figure 3.21, axons from EC layer 2 connect directly into the dentate gyrus (DG), which feeds into CA3 and then on into CA1. As detailed by Rolls and Treves (1998), DG performs pattern sparsification via abundant lateral inhibition: neurons compete with one another to fire, with the active neurons loosely representing principal components. These sparse patterns then enter CA3 via (relatively) direct lines from DG to CA3 pyramidal cells: the fanout is approximately only 1:12 (Andersen et al. 2007), which is quite low for interpopulation connections. CA3 has very high recurrence—the most of any brain region; each CA3 pyramidal connects with about 5 percent of the others via excitatory synapses (Rolls and Treves 1998; Kandel, Schwartz, and Jessell 2000). Thus, CA3 seems to act as a pattern-completing area; and when those patterns have significant temporal scope (as most patterns at higher, slower levels do), pattern completion manifests prediction. In this way, CA3 appears to function as a fill-in-the-blank station for episodic memories (i.e., those of sequences of experiences) cued by a few key hints from DG.

For predictive coding, the pivotal area is CA1, where direct inputs from EC (layer 3) meet the sequence predictions from CA3 (that originated in layer 2 of EC). As shown in figure 3.22, these signal lines meet the dendrites of CA1 pyramidal cells at different locations. The direct lines from EC synapse distally, in a layer containing very little

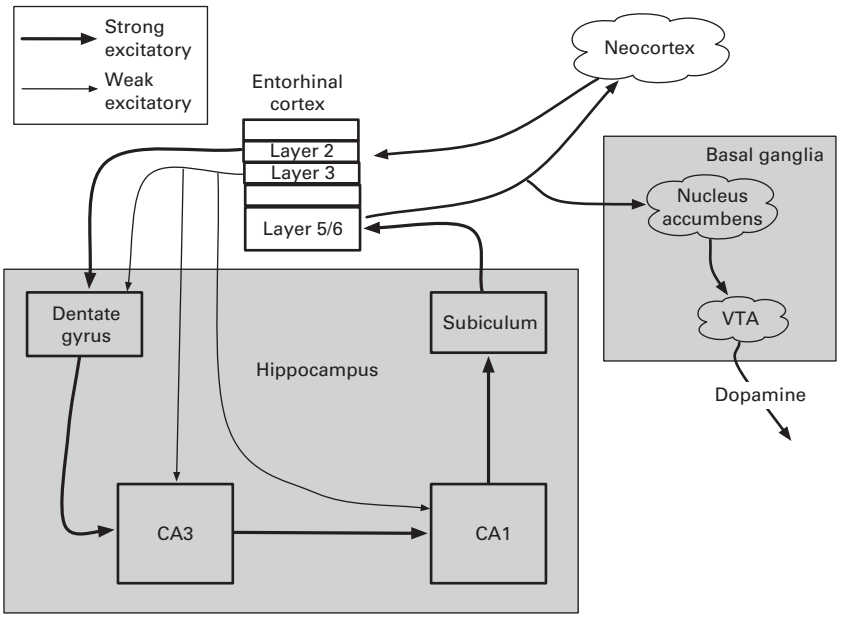


Figure 3.21
Basic topology of the hippocampus (HC), its gateway region (the entorhinal cortex, EC), and other primary sources and targets of HC signals.

inhibition. Hence, these *reality* signals excite the pyramidals. Conversely, the CA3 inputs arrive proximally, near the somas of CA1 pyramidals, in an area very dense with inhibitory interneurons. Thus, although the link between CA3 and CA1 along the Schaffer collaterals is strongly excitatory (Kandel, Schwartz, and Jessell 2000), the net effect may ultimately be inhibitory due to the ensuing interneuron activity. This has led several researchers to propose CA1 as the comparator region for predictions (from CA3) and reality (from EC) (Ouden, Kok, and Lange 2012; Lisman and Grace 2005), with the resulting prediction error representing the level of *novelty* in the current input signal from EC to HC.

As shown in figures 3.21 and 3.23, output from CA1 eventually returns to layers 5/6 of EC on its way back to the neocortex. The novelty signal thus contributes to an abstract prediction of its own (of an expected sequence of upcoming experiences), in much the same way that upward-traveling prediction errors in cortical columns pass through transformative connections before returning downward as feedback predictions. Perhaps more importantly, the novelty signal also reaches the ventral tegmental area (VTA, upper right corner of figure 3.21), which broadcasts dopamine when sufficiently stimulated, and this neuromodulator then stimulates learning.

In summary, those prediction errors that resist being *explained away* by feedback predictions will work their way up the cortical hierarchy to EC, as proposed by Hawkins (2004). The hippocampus then gets one last chance to reconcile the information with its archive of abstract sequences (recalled via activity in DG and CA3). Any remaining error constitutes novelty, which the brain will then try to learn: the neural assemblies that were recently active in the neocortex will be encouraged, by dopamine from VTA, to bond. This learning process continues during sleep, when interaction between HC and neocortex remains strong.

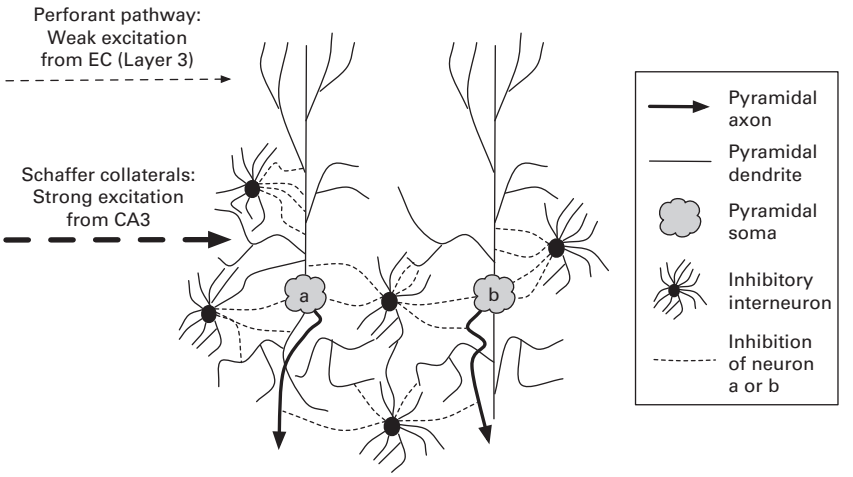


Figure 3.22
General cytology of area CA1 of the hippocampus based on diagrams and descriptions in Andersen et al. (2007). Pyramidal cells in CA1 receive weak excitatory inputs from the entorhinal cortex via distal dendrites, which reside in a layer with very few inhibitory interneurons. Conversely, strong excitation from CA3 via the Schaffer collaterals enters much closer to the soma (a and b), in layers with a much higher density of interneurons of many types, but all inhibitory.

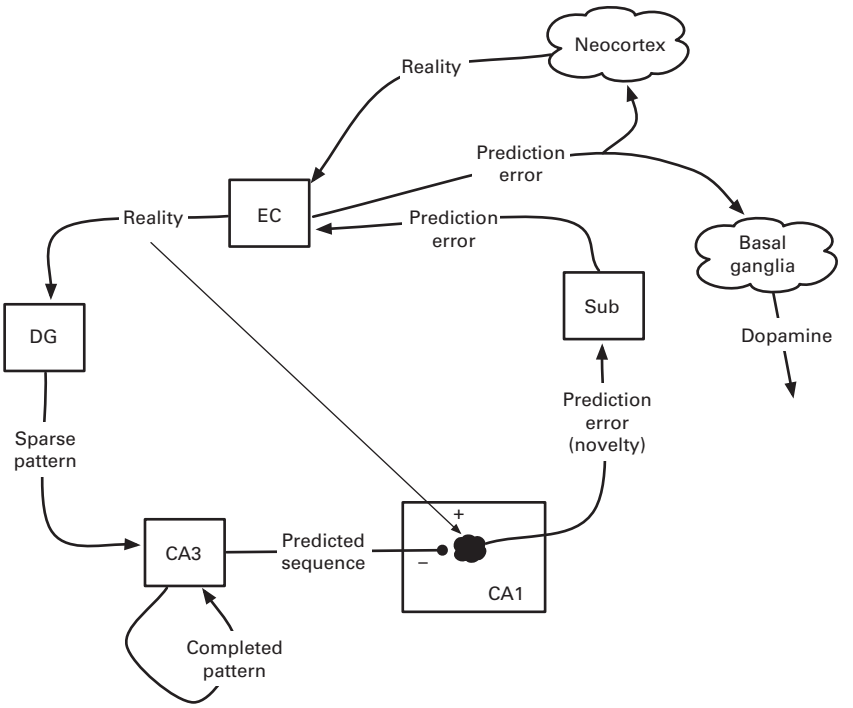


Figure 3.23
Functional diagram of the hippocampal formation as a model of predictive coding. EC = Entorhinal Cortex, DG = Dentate Gyrus, Sub = Subiculum. As hypothesized by several neuroscientists (Ouden, Kok, and Lange 2012; Lisman and Grace 2005), area CA1 acts as a comparator of a prediction from CA3 and reality signal from EC. This relies on the assumption that the excitatory Schaffer collateral pathway from CA3 to CA1 synapses on many of the inhibitory interneurons near the soma and proximal dendrites of CA1 pyramidal cells.

As a domestic example, when my doorbell rings, I immediately begin to conjure up predictions as to the imminent visit. The current context, for example, Halloween night, combines with the doorbell information in driving CA3 to dredge up a memory of trick-or-treaters, which constitutes my expectation (and may cause me to rummage through the house for candy and a makeshift costume). However, when I open the door to find my son, who left his house key at the gym, the deviation of that reality (sent directly to CA1 from EC) to the CA3-generated expectation startles me and may lead to changes of my Halloween memories; twenty years later, I will still tease my son about the time he forgot his key on beggar's night and was met at the door by Socrates and a handful of miniature candy bars.

3.4.2 Conceptual Embedding in the Hippocampus

In 2005, the research group led by May Britt and Edvard Moser discovered *grid cells* in an area of the hippocampal gateway known as the medial entorhinal cortex (MEC) (Hafting et al. 2005). The discovery earned them the 2014 Nobel Prize in medicine, an honor that they shared with their mentor, John O'Keefe, who discovered *place cells* in the hippocampus three decades earlier (O'Keefe and Dostrovsky 1971). The differences between grid and place cells, along with their interactions, provide an interesting backdrop for navigation, prediction, and intelligence in general.

Place cells are neurons in hippocampal regions such as CA3 and CA1 that tend to fire only when the organism resides in (or approaches) a particular location (Andersen et al. 2007). Hence, each place cell constitutes a detector for some fairly localized spot in the environment. In contrast, grid cells have the fascinating property of detecting all spots at the corners (and center) of hexagonal patterns laid out across an environment. Thus, a particular grid cell systematically cycles through active and inactive phases as the animal (to date, typically a rodent) moves about its environment. As depicted in figure 3.24, if an experimenter marks the spots where that cell fires on a map of the environment, a pattern of equilateral triangles arises; and together, these form hexagons: the receptive fields of grid cells are hexagonal grids.

Figure 3.24 also illustrates that as the animal moves in a fixed direction, the pattern of grid-cell activity should essentially follow a cyclic pattern as the corners of overlapping triangles are visited. As explained more thoroughly by the Mosers and colleagues (2014), inputs to grid cells from both head-direction and velocity cells are believed to push activation patterns around MEC, producing cyclic activity bumps among grid cells with similar receptive fields. The faster the animal moves, the more rapidly does the bump. Interestingly, the Moser Lab has also found evidence of a special cluster of MEC cells that represent context-free velocity (Kropff et al. 2015), and they too appear to be predictive: their activity correlates better with future than with past or present speed.

The precise purpose(s) of grid cells and their hexagonal receptive fields have yet to be proven, but most believe that they provide a form of mental global positioning system (GPS) when combined with place cells. Basically, when an animal resides in a given spot, S, and moves in a known direction with a known speed, then it can estimate its new location (S*) without receiving much sensory feedback, for example, in the dark. In effect, the animal makes a *prediction* via a process of dead reckoning wherein grid cells support sensory-feedback-free transitions between place cells.

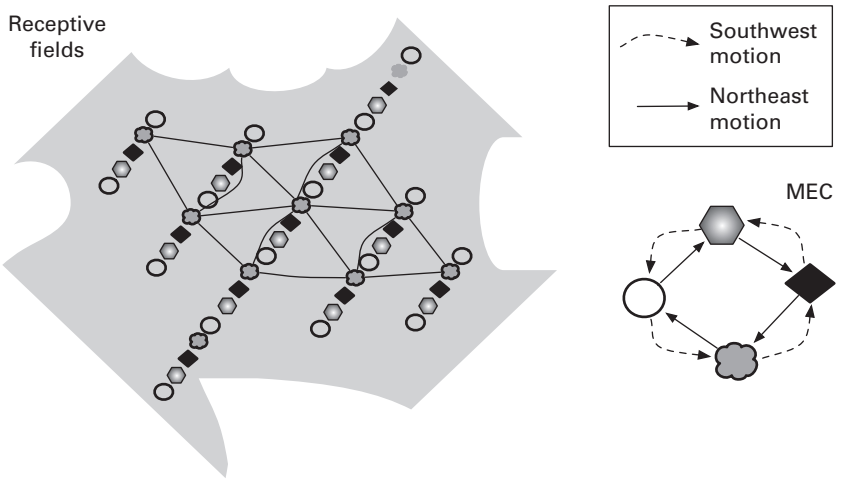


Figure 3.24
The basic behavior of grid cells. (Left) An environment (shaded background) labeled with small shapes denoting the grid cell that fires when the organism resides in that particular location. The hexagonal pattern depicts a cluster of places (i.e., the corner points of each triangle) where the neuron, represented by a shaded, flower-shaped pattern, will fire, aka that neuron's *receptive field*. Similar hexagonal receptive fields exist for each of the four neurons. (Right) Relationships between four grid cells for achieving the receptive fields on the left. When the agent moves in a northeast (southwest) direction, the solid (dashed) arrows portray the sequence of grid-cell stimulation: the activation of any grid cell combines with sensorimotor inputs to stimulate its neighbor. When neighboring grid cells have similar receptive fields, the cyclic activation pattern is that of a moving bump or blob in a multidimensional neural space.

Underlying this dead reckoning is the ability to triangulate a location using multiple grid cells. MEC houses grid cells whose receptive fields vary in spatial frequency, phase, and rotation (Moser et al. 2014). Thus, as shown in figure 3.25, the receptive fields of different cells may intersect at only a few points, and the more grid cells involved, the fewer points of mutual activation.

The information encoded in the firing of a single grid cell is rather diffuse, indicating only that the animal resides near one of the many triangle corners in its hexagonal grid. But when two or more grid cells (of different frequency, phase, and/or rotation) fire, this helps pinpoint the actual location. In the cartoon example of figure 3.25, it takes only two grid cells to reduce the number of possible locations to three, and complete disambiguation results from the addition of a third grid cell. In reality, this reduction of possibilities may require four or more grid cells, but the basic process seems quite plausible.

Thus, combinations of active grid cells may indicate unique locations, that is, *places*, and connections from MEC to the interior of the hippocampus (areas CA3 and CA1) may constitute detector circuits for those places. As shown in figure 3.26, both place cells, P_a and P_b, have incoming connections from a set of three different grid cells. When one of these grid triples fires, so too does the corresponding place cell. The figure omits many details, including the recurrent pathways that connect CA3 and CA1 back to MEC (via the subiculum) (Moser et al. 2014; Andersen et al. 2007). This feedback could enable an active place cell to excite a group of related grid cells: those that triangulate that place cell.

The animal then makes predictions of places, either those in the future or in the present but invisible due to limited or delayed sensory input, by running these circuits in both directions.

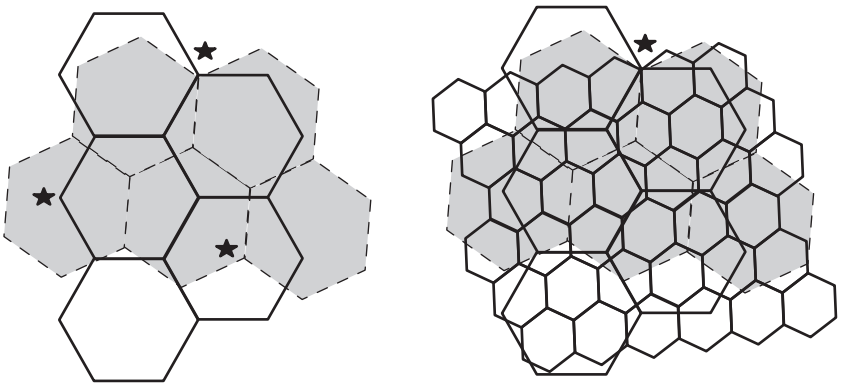


Figure 3.25
Triangulation of environmental locations using grid cells. (Left) Each set of hexagons (one with thick borders and the other with dashed borders and gray interior) represents the receptive field of a different grid cell. Starred points are those where the fields intersect, either at the corners or centers (not drawn) of hexagons. (Right) Addition of a third grid field (in this case of higher frequency, i.e., smaller hexagons) reduces the number of three-point intersections to the single starred location.

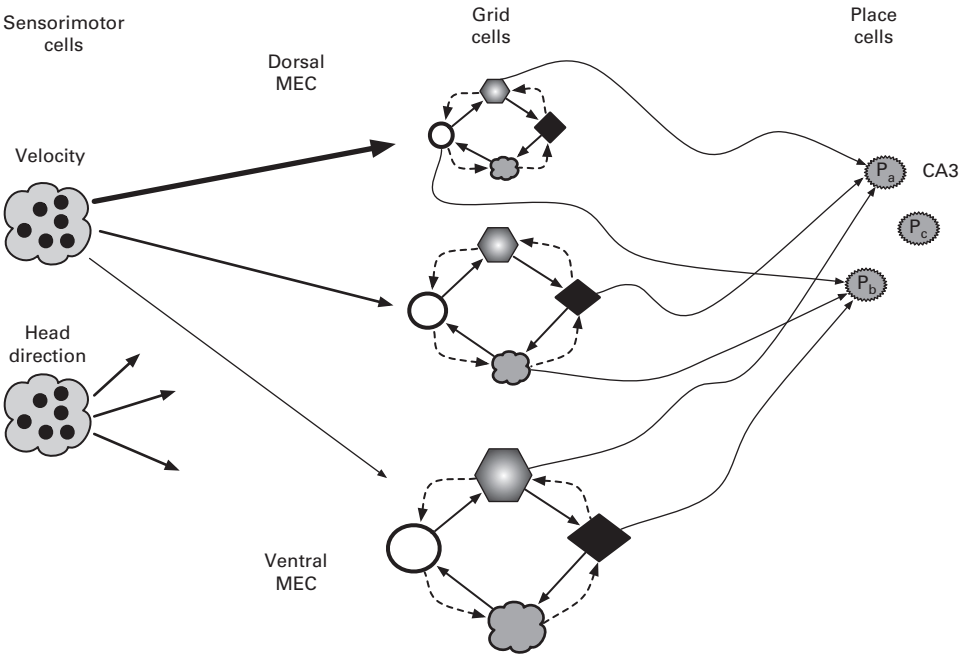


Figure 3.26
Interactions among sensorimotor information, grid cells, and place cells. In moving from the dorsal to ventral MEC, grid cells have lower-frequency (higher amplitude) receptive fields denoted by larger cell icons (Moser et al. 2014; Andersen et al. 2007). For any given frequency and orientation, the grid cells representing different spatial phases (offsets) might interact via cyclic excitation (as denoted by the three bidirectional cycles). As a simplified interpretation, the active head-direction cells could *select* the appropriate cycle at each frequency level, while velocity inputs would drive the activity bump around those focal cycles. The thickness of arrows emanating from the velocity group indicate strength of influence (e.g., synaptic strengths) such that a given velocity should drive a high-frequency grid cycle harder (faster) than a low-frequency loop.

Initially, the agent surmises its current location via sensory cues, such as spatial landmarks, olfactory or auditory signals, and so on. Detectors for these sensory data also serve as inputs to the hippocampus, indirectly via MEC and other regions of the entorhinal cortex. Hence, these detectors can activate place cells (P*), which, in turn, can *reset* the grid system such that the triangulators (G*) of all active place cells also begin firing.

Next, the animal moves but without receiving sufficient sensory information, either due to sensory deprivation (e.g., darkness) or owing to the relatively long delays in sensory processing compared to those of action. However, the proprioceptive input enables the agent to estimate its velocity and egocentric head direction, and this information, in turn, can push the activity bumps associated with G* through several steps of their associated cycles. The final locations of those bumps then represent a set of active grid cells that map to a new set of place cells that represent the predicted location. The later arrival of external sensory information may then confirm or refute that prediction, but even if incorrect, the prediction may provide advantages over the completely naive state.

A major question in cognitive neuroscience is whether or not the grid-place-cell network could be employed for tasks other than navigation (Bellmund et al. 2018). The hippocampus is already known to be critical for long-term memory formation (Andersen et al. 2007; Kandel, Schwartz, and Jessell 2000). The co-location of a navigational and memory system gives neural support to memory-enhancing techniques that involve picturing the sequence of memory items around one's living room or along a familiar trail. But what about other aspects of cognition, such as planning or reasoning in general?

One possibility involves the conceptual spaces (Gärdenfors 2000) discussed briefly in chapter 2 (and related to the hippocampus according to the Moser group (Bellmund et al. 2018)). Might grid cells provide a mechanism for systematically moving about such spaces, with many possible actions (and their intensities) replacing head-direction and velocity cells, respectively? The key gradient in this scenario is that of grid-bump movement with respect to different actions, and such derivative information might be manifest in the synaptic strengths of connections from premotor areas to the entorhinal cortex.

For example, consider the planning that a coach might do for a sports team. Typical spectra of prominence include (a) the health and physical condition of the team, (b) the confidence that the players have in their own physical and mental preparedness, and (c) the coherence and unity that the players exhibit in their style of play and attitudes toward one another. In reasoning about these factors and their interactions, each of these three spectra might link to a particular set of grid cells, as would neurons representing particular actions, such as those of the premotor cortex. Then, reasoning about the consequences of one such action (such as the grueling group sprints that most athletes detest) with respect to a spectrum (such as that of health-and-conditioning) would *drive the bump* along the relevant subnetwork of MEC.

Expecting these spectra to wrap around to form cycles (or a torus in a multidimensional context) may be unrealistic in some cases, but all three mentioned above have clear cyclic tendencies if driven by certain actions. Physical condition, for example, normally improves with exercise, but an excess can quickly produce injuries and, essentially, a wraparound from *excellent* to *poor*. The same holds for the confidence spectrum, wherein players can become increasingly positive about their abilities as they tackle more challenges, but one bad experience against a dominant opponent (i.e., one challenge too many) can undo a lot of

progress if not properly addressed by the coach during and after the defeat. Finally, coherence may gradually improve with repetition of systematic plays among a stable group of players, but adding in new plays or players too quickly can move the needle straight back to chaos.

The coach's predictive reasoning might then go as follows. After assessing the current points on each of these spectra at the start of a season, the coach comes up with a training plan (a set of actions) designed to improve, or at least maintain, each of them. The current state would correspond to a location in an abstract space, and thus one or more related place cells (P*) in the hippocampus. These place cells would invoke a set of grid cells (G*) that best support them. Next, in contemplating the training plan, the coach invokes several potential actions (A*), including physical exercises and drills, team meetings, and the scheduling of early-season practice opponents.

Then, G* and A* interact to predict the team's future state. First, activation of A*-correlated neurons drives various MEC bumps from the G* state to a new grid state, G**. Next, G** maps to a new place-cell group, P**, which corresponds to the predicted state of the team. This *turn-the-MEC-crank* model of planning simply exapts the entorhinal and hippocampal predictive machinery that originally evolved for a purely spatiotemporal task, navigation. It relies heavily on an ability of the brain to map conceptual spaces (i.e., spectra) to grid cells and to incorporate a gradient-based understanding of the causal relationships between agent' actions and translations within these spectra.

In short, learning to make good predictions, of many varieties, may involve tuning of the synapses between the neo- and entorhinal cortices, and the hippocampus proper. The human use of spatiotemporal analogies and metaphors is well documented in mathematics (Lakoff and Nunez 2000) and in numerous other areas (Lakoff and Johnson 1980). This provides psychological evidence for a neural theory linking many cognitive abilities (including prediction) to the navigational apparatus of the hippocampal system. Although nothing has been proven, the possibility comes up frequently in research articles on grid and place cells.

3.5 Gradients of Predictions in the Basal Ganglia

The neural mechanisms for prediction and gradient calculation also come together in the basal ganglia, shown in figure 3.27. Roughly speaking, the basal ganglia receives inputs from the cortex via the striatum, whose neurons have firing properties amenable to *context detection*: they require a significant amount of cortical input in order to fire, so those cortical firing patterns constitute a context, which the basal ganglia then maps to two outputs: an action and an evaluation. Actions are (clearly) critical for proper behavior, while evaluations motivate the neural modifications that underlie learning.

The basal ganglia's action-evaluation separation closely matches a fundamental paradigm of reinforcement learning (RL) known as the *actor-critic model* (Sutton and Barto 2018), wherein an actor module handles action selection while its counterpart, the critic, handles evaluation. In RL, communication between the two is restricted to an error term (computed by the critic) that amounts to a temporal prediction gradient: the difference between predictions made at times t+1 and t.

Focusing on the critic segment of figure 3.27 (right), several pathways lead from the striatum to the substantia nigra pars compacta (SNc), which, by many accounts (Barto

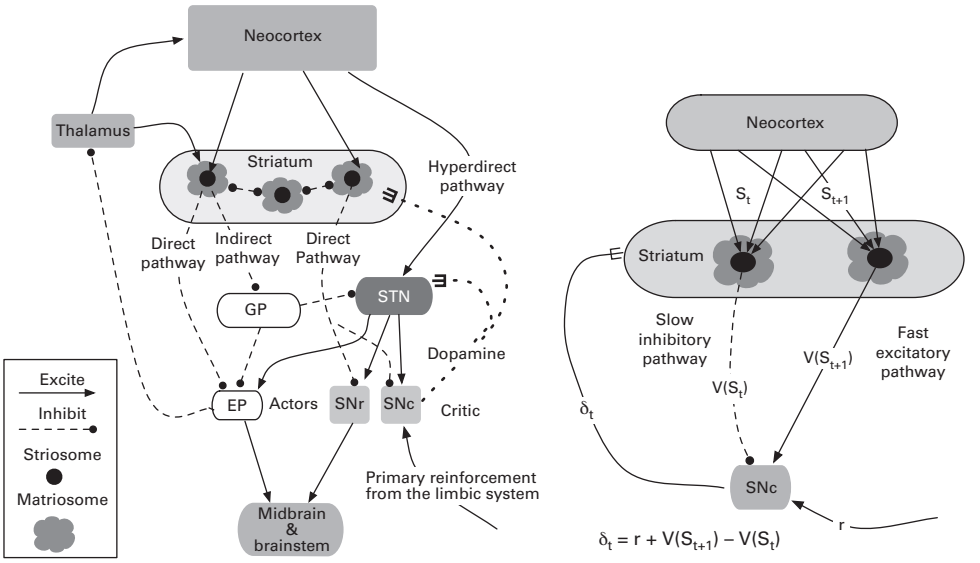


Figure 3.27
(Left) General functional anatomy of the mammalian basal ganglia. (Right) Computation of temporal-difference error (δ_t) by the SNc of the basal ganglia.

1995; Houk, Davis, and Beiser 1995; Graybiel and Saka 2004) constitutes the critic's core. Although the neuroscientific accounts vary, many models (Graybiel and Saka 2004; Prescott, Gurney, and Redgrave 2003) agree on the presence of both (a) fast excitatory connections from the neocortex and striatum to the SNc, and (b) slower inhibitory links between those same regions. Finally, one well-known input to the SNc comes from the amygdala, the center of the brain's emotional response (LeDoux 2002); it activates whenever the body experiences pleasure or pain.

Taken together, these three main pathways to the SNc (summarized in figure 3.27, right) form an ideal manifestation of RL's central prediction gradient, known as the *temporal difference error* (TDE). Essentially, TDE represents the difference between (a) the estimated value ($V(s_t)$) of the state/context(s) experienced by a system at time t, and (b) the combination of another estimate ($V(s_{t+1})$) for the system's next state, and any reward or penalty (r) incurred between times t and t+1. In RL, this is commonly expressed as

$$\delta_t = V(s_{t+1}) + r - V(s_t) \tag{3.2}$$

TDE (δ_t) is then combined with a learning rate (α) to update the estimate $V(s_t)$:

$$V(s_t) \leftarrow V(s_t) + \alpha\delta_t \tag{3.3}$$

This embodies *bootstrapping*, wherein an estimate made at time t is improved by one made at time t+1. Why should estimates made at t+1 be any more accurate than those at t? The answer revolves around the formal definition of these estimates: in RL theory, $V(s_t)$ embodies a *prediction* as to the amount of reward that the agent will accrue from time t until the end of that particular problem-solving attempt (formally known as an *episode*), often denoted as time T. It therefore makes sense that a prediction made at time t+1 and involving

a fixed time horizon (T) should be marginally more accurate than one made at time t. In addition, the traversal from s_t to s_{t+1} involves concrete feedback (reward or punishment), r, which can only enhance the realism of $V(s_t)$. In short, the derivative of the prediction (supplemented with the immediate reward) provides the basis for an improvement of the prediction.

Returning to the basal ganglia, the analog of $V(s_t)$ is the activity level of the striatal neurons that have been tuned to detect s_t, which, via the fast excitatory pathway, correlates with the strength of positive inputs to SNc. Crucially, the SNc controls the diffuse secretion of the neuromodulator dopamine, whose presence enhances the synaptic modifications that enable learning; and the striatum receives a substantial amount of this dopamine, thereby affecting the synapses from cortex to striatum. Of special importance here is the well-documented fact (Schultz et al. 1992) that dopamine secretion results from a mismatch between expectations and reality, but not from the mere presence of a reward signal from the amygdala. In other words, when that reward is *expected*, the SNc remains inactive. In terms of the incoming connections to SNc in figure 3.27, a high value of r excites SNc only if not counterbalanced by a strong inhibitory signal from the striatum, where that signal represents $V(s_t)$, a prediction of reward.

Furthermore, SNc stimulation can occur in the absence of an amygdalar impulse. When the magnitude of the excitatory input from the fast pathway (which conveys $V(s_{t+1})$) exceeds that of the delayed inhibitory signal (for $V(s_t)$), this difference between two predictions can also activate SNc, resulting in dopamine and learning. In psychological terms, this difference represents a heightened anticipation of reward. In the context of this chapter, this difference signifies a positive prediction gradient that leads to learning, which works to reduce that gradient by increasing $V(s_t)$. As shown in the figure, the activity level of the SNc embodies the TD error of RL, via its relationship to $V(s_t)$, $V(s_{t+1})$ and r.

One final question involves timing. The SNc computes a form of TD error at time t+1 (or slightly thereafter) based on a fast excitatory value signal pertaining to s_{t+1} plus a delayed inhibitory signal representing $V(s_t)$. So how does the dopamine produced at time t+1 lead to an update of the synapses encoding $V(s_t)$ but not $V(s_{t+1})$? One plausible answer stems from a complex network of chemical interactions involving some inherent latencies (of approximately 100 milliseconds) (Houk, Adams, and Barto 1995; Downing 2009), the details of which are beyond the scope of this book. This intricate neurochemistry ensures that only those synapses (from neocortex to striatum) that have recently (but not too recently) been active are susceptible to the modifications induced by dopamine.

In summary, the basal ganglia calculates a temporal prediction gradient via complex neural circuitry that includes a crucial delayed inhibitory signal. The difference between a fast excitatory and delayed inhibitory signal (both representing the values of states in terms of their *predicted* future reward) combines with a fast excitatory indicator of *immediate* reward to yield a signal whose neurophysiological manifestation is dopamine and whose computational analog is the classic temporal-difference error of reinforcement learning theory. In both fields of study, that signal leads to learning of an improved prediction for the context associated with the delayed inhibitory signal: s_t.

Whereas navigating bacteria and our early levels of neural processing use gradients as the basis for a prediction, in higher levels of the brain, such as the basal ganglia, the key gradients are of the predictions themselves, and these gradients then govern tuning of the

predictive machinery. This general trend continues up into the highest brain regions, where a great many signals probably represent abstractions and expectations rather than reality.

3.6 Procedural versus Declarative Prediction

In an earlier pair of journal articles, I analyzed several brain regions (neocortex, hippocampus, thalamus, cerebellum, and basal ganglia) with respect to procedural versus declarative knowledge (Downing 2007a) and then how these differentially facilitate prediction (Downing 2009). These same cranial regions receive considerable treatment in this and other chapters, although this book will not delve as deeply into those neural circuits as do the two articles. Here, the focus is more on the mathematical and computational components of predictive networks.

However, the distinction between procedural and declarative prediction deserves mention. In the former, an agent may act *as if* it had knowledge of a future state without actually having an internal representation of that expectation. For example, gradient-following bacteria exhibit procedurally predictive behavior. The cerebellum and basal ganglia often carry the *procedural* label, as their numerous parallel cables manifest relatively hardwired (though modifiable) links between contexts and actions. The sheer density of these connections equips the organism with many tunable if-then situation-action rules for survival, and many of these handle the temporal differences required for predictions (as detailed in Downing 2009). But they do not facilitate the formation of stable neural activation patterns suggestive of *representations*. Without stability, a pattern cannot persist long enough to form the basis of attention, which is critical for everything from simple conscious reasoning to the advanced use of symbols and language (Deacon 1998).

The hippocampus and neocortex (alone and in combination with the thalamus) have a different architecture, one lacking parallel lines but replete with recurrent connections; and these often provide the feedback necessary to produce stable activation patterns. When those patterns represent future states, they constitute declarative predictions. Similarly, they can represent declarative goals; and both predictions and goals, when encoded as activation patterns, can serve as inputs to comparators that combine their inverses with representations of reality. These differences can have downstream effects that make goals and predictions functionally significant. Forming and combining declarative representations provides much more computational flexibility than do behaviors embodied solely in situation-action associations. Thus, higher organisms, with more declarative capabilities (evidenced by expanded hippocampal and/or neocortical regions) are capable of more complex problem solving.

3.7 Rampant Expectations

The organisms, simple circuits, and complex brain regions above illustrate key relationships between several core concepts of this book: gradients, sums, predictions, and adaptation. Bacteria employ chemistry to compute gradients, which then influence behavior *as if* the organism has informed expectations about its immediate spatiotemporal future. The nematode worm displays similar implicit (procedurally predictive) behavior, but using basic desensitization mechanisms within its primitive nervous system. These behaviors exhibit

procedural predictivity in the same way that a baseball outfielder *predicts* where a fly ball will land: he gradually moves to the proper location while tracking the ball but presumably has no internal mental representation of the final destination. Similarly, the outfielder uses gradient information (changes to the elevation angle of the ball) to adjust his own velocity and direction of movement. In all of these cases, the gradient enables prediction, albeit implicitly.

Conversely, in the mammalian visual system (and many other brain regions), neighborhood activity-level averages function as predictions of the activity level of any neuron (N), with the difference between N's activation and the prediction constituting a spatial gradient sent up the neural hierarchy. So in this case, the prediction enables gradient calculation, which, at the next level, supports further averaging and prediction. Note that the predictions themselves become more explicit in this context, since they are directly reflected in the total inhibition received by N. Here, the immediate result is not improved overt behavior but more broadband and energy-efficient signal transmission.

In the simple computational models based on Tripp and Eliasmith's (2010) neural motifs for temporal differentiation, explicit predictions stem from the leaky integration of rising temporal gradients. When projected back downward, these predictions normalize the lower layer's state value to form an error term. In chapter 5 on predictive coding, that error term is shown to serve as the main ascending signal. These models indicate that the interactions between gradients and integrative predictions support neural hierarchies in which higher layers can run at much slower speeds while still providing coarse, but reasonably accurate, predictions of lower-level behavior.

These simple circuits, whether alone or in repetitive layers, provide relatively simple, generic mechanisms for prediction that could potentially exist in many parts of the brain, particularly the neocortex, either as large fields of predictors or as small islands of expectation production. In contrast, several complex, heterogeneous brain areas appear to house potent predictive machinery of very specialized design, as seen in the cerebellum, hippocampus, and basal ganglia.

The convergence of prediction and control seems particularly evident in the cerebellum, a pivotal area for the timing and coordination of both motor and cognitive activities. Although this region is often modeled as an adaptive controller, it does not cleanly decompose into neural modules that map to regulator components. Instead, the numerous granular cells and emanating parallel fibers appear to transmit goals, predictions, and sensory reality, which then combine at comparators in the Purkinje cells to yield prediction errors, which then determine control signals sent to actuators, whose efference copies then provide predictions that feed back into the granular cells. Interactions of this control loop with a teaching signal from the inferior olive provide the brain's best example of adaptation via supervised learning (Doya 1999).

The hippocampus is particularly intriguing for its predictive potential. First, area CA1 has been proposed by neuroscientists as a comparator of reality signals—coming directly from the entorhinal cortex (EC)—and (slightly delayed) predictions based on pattern sparsification in area DG followed by pattern-completion of memories in area CA3. These memories serve as more abstract, slower-time-scale representations, whose comparison to more-immediate sensory reality indicates the level of surprise associated with the current situation. In addition, the interaction between the EC's grid cells and the hippocampal place

cells has very strong implications for prediction, with the former updating the latter as to current and future locations of the agent in the absence of sufficient teleosensory information (e.g., sight, sound, smell) but based on transitions in collections of grid-cell networks as driven by proprioceptive velocity and orientation signals. Here, gradient knowledge, of how changes in position (i.e., velocity) affect activity-bump movement in grid cells, directly supports navigation by dead reckoning; and this same apparatus may be co-opted for numerous other cognitive activities (Bellmund et al. 2018).

Finally, the architecture and dynamics of the basal ganglia couple gradients, prediction, and adaptation. Employing signal-delay circuitry (similar to Tripp and Eliasmith's motifs) in combination with immediate reinforcement information from the amygdala, the basal ganglia compute predictions (of future reinforcement) at two adjacent time points, the difference of which manifests a temporal prediction gradient, which essentially represents *surprise*. Adaptation, via dopamine-enhanced synaptic modification to basal gangliar afferents, is then driven by the level of surprise.

As this chapter indicates, there are many neural structures capable of generating predictions. Some employ gradients and integrals in simple networks, while others exploit these same basic components in circuits resembling PID or adaptive controllers, thus blurring the boundaries between prediction and control. Reconciling these functional models with actual neural anatomy and physiology is always a speculative affair, but the entire field of computational neuroscience is built on a wide range of theories having supporting, but hardly confirming, evidence. But if prediction is indeed one of the brain's primary functions, then some of these predictive interpretations of contemporary neuroscience deserve further consideration and exploration.

The MIT Press would like to thank the anonymous peer reviewers who provided comments on drafts of this book. The generous work of academic experts is essential for establishing the authority and quality of our publications. We acknowledge with gratitude the contributions of these otherwise uncredited readers.

This book was set in Times New Roman by Westchester Publishing Services.

Library of Congress Cataloging-in-Publication Data

Names: Downing, Keith L., author.
Title: Gradient expectations : structure, origins, and synthesis of predictive neural networks / Keith L. Downing.
Description: [Cambridge, Massachusetts] : The MIT Press, [2023] | Includes bibliographical references and index.
Identifiers: LCCN 2022037237 (print) | LCCN 2022037238 (ebook) | ISBN 9780262545617 (paperback) | ISBN 9780262374682 (epub) | ISBN 9780262374675 (pdf)
Subjects: LCSH: Deep learning (Machine learning) | Neural networks (Computer science) | Conjugate gradient methods.
Classification: LCC Q325.73 .D88 2023 (print) | LCC Q325.73 (ebook) | DDC 006.3/2—dc23/eng20230302
LC record available at https://lccn.loc.gov/2022037237
LC ebook record available at https://lccn.loc.gov/2022037238