CONTENTS

Acknowledgments xi
Prologue xiii

Introduction 1
The Perspective of Neurobiological Information 4
The Perspective of Algorithmic Information 5
A Shared Perspective 7
The Ten Seminars 11
On Common Ground 24

The Present and the Past 26
The First Discussion: On Communication 26
The Historical Seminar: The Deeply Engrained Worship of Tidy-Looking Dichotomies 36

1 ALGORITHMIC GROWTH 81

1.1 Information? What Information? 83
The Second Discussion: On Complexity 83
Seminar 2: From Algorithmic Growth to Endpoint Information 91

1.2 Noise and Relevant Information 112
The Third Discussion: On Apple Trees and the Immune System 112
Seminar 3: From Randomness to Precision 120
1.3 Autonomous Agents and Local Rules
   The Fourth Discussion: On Filopodia and Soccer Games 139
   Seminar 4: From Local Rules to Robustness 146

2 OF PLAYERS AND RULES 161

2.1 The Benzer Paradox 163
   The Fifth Discussion: On the Genetic Encoding of Behavior 163
   Seminar 5: From Molecular Mechanisms to Evolutionary Programming 170

2.2 The Molecules That Could 186
   The Sixth Discussion: On Guidance Cues and Target Recognition 186
   Seminar 6: From Chemoaffinity to the Virtues of Permissiveness 192

2.3 The Levels Problem 211
   The Seventh Discussion: On Context 211
   Seminar 7: From Genes to Cells to Circuits 217

3 BRAIN DEVELOPMENT AND ARTIFICIAL INTELLIGENCE 237

3.1 You Are Your History 239
   The Eighth Discussion: On Development and the Long Reach of the Past 239
   Seminar 8: From Development to Function 245

3.2 Self-Assembly versus “Build First, Train Later” 262
   The Ninth Discussion: On the Growth of Artificial Neural Networks 262
   Seminar 9: From Algorithmic Growth to Artificial Intelligence 267
## CONTENTS ix

3.3 Final Frontiers: Beloved Beliefs and the AI-Brain Interface 287  
*The Tenth Discussion: On Connecting the Brain and AI* 287  
*Seminar 10: From Cognitive Bias to Whole Brain Emulation* 294  
Epilogue 312  

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Glossary</td>
<td>317</td>
</tr>
<tr>
<td>References</td>
<td>329</td>
</tr>
<tr>
<td>Index</td>
<td>351</td>
</tr>
</tbody>
</table>
There are easier things to make than a brain. Driven by the promise and resources of biomedical research, developmental neurobiologists are trying to understand how it is done. Driven by the promise and advances of computer technology, researchers in artificial intelligence (AI) are trying to create one. Both are fields of contemporary research in search of the principles that can generate an intelligent system, a thing that can predict and decide, and maybe understand or feel something. In both developmental neurobiology and AI based on artificial neural networks (ANNs), scientists study how such abilities are encoded in networks of interconnected components. The components are nerve cells, or neurons, in biological brains. In AI, the term neuron has been readily adopted to describe interconnected signaling components, looking back on some 70 years of ANN research. Yet, to what extent the biological analogy is useful for AI research has been a matter of debate throughout the decades. It is a question of how much biological detail is relevant and needed, a question of the type of information necessary to make a functional network. The information problem underlies both fields. What type of information is necessary to wire a brain? What do biologists mean when they say something is “encoded by genes,” and how is genetic information transformed into a brain? And finally, to what extent is the same type of information required to wire up biological brains or to create artificial intelligence?

This book is about the information problem and how information unfolds to generate functional neural networks. In the case of biological
brains, prior to learning, the information for developmental growth is encoded in the genome. Yet, there are no chapters about brain regions or their connectivity to read in the genome. In fact, compared to the information necessary to describe every detail necessary to make a functioning brain, there is rather little information available in the genome. Growth requires genetic information plus time and energy. Development happens in steps that occur in space and time in an ordered fashion. The outcome is a system that would require more information to describe than was needed to start its growth. By contrast, most ANNs do not grow. Typically, an artificial network with initially random connections learns from data input in a process that is reminiscent of how biological brains learn. This process also requires time and energy. Learning also occurs in steps, and the order of these steps matters. There are important similarities and differences between these stepwise, time- and energy-consuming processes. The current hope for AI based on ANNs is that the learning process is sufficient and that a developmental process analogous to biological brains can therefore be omitted. Remarkably, there was a time in neurobiology research almost a hundred years ago when scientists felt much the same about the brain itself. It was inconceivable where the information for wiring should come from other than through learning. The idea was that, just like ANNs today, the brain must initially be wired rather randomly, and subsequent learning makes use of its plasticity.¹ But if this were so, how could, say, a monarch butterfly be born with the ability to follow thousands of miles of a migration route that it has never seen before?

As temperatures drop in the fall in North America, millions of monarch butterflies migrate for up to 3,000 miles to overwinter in Mexico. Remarkably, millions of butterflies distributed over close to 3 million square miles in the north all target only a few overwintering sites that cover less than a single square mile. Many theories have been put forth as to how a butterfly could do this.²,³ Similarly remarkable, an individual sea turtle will return over thousands of miles to the very beach where it was born—many years later. We do not know how sea turtles do it, but it is conceivable that they had learned and can remember something
about a place where they had once been before. This is where the story of the monarch butterfly turns from remarkable to downright unbelievable. The butterflies that started out in the north will overwinter in the south until temperatures rise next spring. They then start flying north again, but only a few hundred miles. At different places in the southern United States they stop, mate, lay eggs and die. A new generation of monarchs picks up the trail north, but again only for a few hundred miles. It usually takes 3–5 generations for a full round trip. By the time temperatures drop again in the fall in North America, a monarch butterfly is about to embark on the 3,000-mile trip south to a precise location that was last visited by its great-great-grandfather. Where is this information coming from?

The currently almost exclusive focus of AI on ANNs is a highly successful, but recent development. It followed several decades during which AI and machine learning focused on formal, symbol-processing logic approaches, rather than the somewhat enigmatic neural networks. For most of its history, AI researchers tried to avoid the complexities and messiness of biological systems altogether. How does information about the role of a gene for a neuronal membrane protein help to program an intelligent system? The history of AI is a history of trying to avoid unnecessary biological detail in trying to create something that so far only exists in biology. The observation begs the question what information can safely be deemed “unnecessary.” To address this question, we need to look at biological and artificial brain development from the information perspective. An assumption and hope of AI research has long been that there is a shortcut to creating intelligent systems. We may not yet know what shortcuts work best, but it seems a good idea to at least know exactly what it is we are trying to leave out in attempts to create nonbiological brains. My hope is that an understanding of the way information is encoded and transformed during the making of biological brains proves useful in the discussion what can and cannot be shortcut in the making of AI. This is the story of a neurobiologist tracking down that information.
The Perspective of Neurobiological Information

The biological brain is a complicated network of connections, wired to make intelligent predictions. Common analogies for brain wiring include circuit diagrams of modern microprocessors, the electrical wiring installations in skyscrapers or the logistics of transportation networks in big cities. How are such connections made during brain development? You can imagine yourself trying to make a connection by navigating the intricate network of city streets. Except, you won’t get far, at least not if you are trying to understand brain development. There is a problem with that picture, and it is this: Where do the streets come from? Most connections in the brain are not made by navigating existing streets, but by navigating streets under construction. For the picture to make sense, you would have to navigate at the time the city is growing, adding street by street, removing and modifying old ones in the process, all the while traffic is a part of city life. The map changes just as you are changing your position in it, and you will only ever arrive if the map changes in interaction with your own movements in it. The development of brain wiring is a story of self-assembly, not a global positioning system (GPS).

When engineers design the electrical wiring in a building or a computer microchip, they have the final product in mind. We make blueprints to understand and build engineered systems with precise outcomes. A blueprint shows a picture of the final product, the endpoint. A blueprint also contains all the information needed to build that product. It largely doesn’t matter in what order the pieces are put in, as long as everything is in place when you flip the on switch. But there is no blueprint for brain connectivity in the genes. There is also no such information coming from the environment. If neither the genes nor the environment contain endpoint information of connectivity, what kind of information do they contribute?

Genetic information allows brains to grow. Development progresses in time and requires energy. Step by step, the developing brain finds itself in changing configurations. Each configuration serves as a new basis for the next step in the growth process. At each step, bits of the
genome are activated to produce gene products that themselves change what parts of the genome will be activated next—a continuous feedback process between the genome and its products. A specific step may not have been possible before and may not be possible ever again. As growth continues, step by step, new states of organization are reached. Rather than dealing with endpoint information, the information to build the brain unfolds with time. Remarkably, there may be no other way to read the genetic information than to run the program. This is not a trivial statement to make, and it will take some explaining. If there is no way to read the genetic code other than running it, then we are principally unable to predict exact outcomes with any analytical method of the code. We can simulate it all right, but the result would not have been predictable in any way other than actually running the whole simulation. The information is in the genes, but it cannot be read like a blueprint. It really is a very different type of information that requires time and energy to unfold.

The Perspective of Algorithmic Information

Scientists in nonbiological fields are more familiar with this type of information. There is a simple game, where you draw lines of X’s or O’s (or black dots versus blanks) based on simple rules that produce remarkable patterns. Imagine a single X in a row of an infinite number of O’s and a simple rule that determines for each triplet of X’s and O’s whether there is an X or an O in the next row. To find out the next line, you read the first three characters, write the output X or O under the center of the triplet below, then move one character and do it again for the next partially overlapping triplet. One rule, called rule 110, looks innocently enough like this:

Triplet in previous row: XXX XXO XOX XOO OXX OXO OOX OOO
..determines in next row:  O   X   X   O   X   X   X   O

For example, starting with one X: .. OOOOXXXXOOOOO..
will lead to the next row: .. OOOOXXXXOOOOO.
Repeating this process again and again, using each previous line to apply the rule and write the next one below, will create a two-dimensional pattern (you will find the result in figure 2.3 on page 96). The repeated application of defined rules is an iteration. A ruleset that uses the output of each preceding step as the input of the next step defines an algorithm. The two-dimensional pattern is the outcome of algorithmic growth based on the iterative application of simple rules. But what does this game have to do with brain development? Shockingly, for the simple rule shown above, the two-dimensional pattern turns out to be so surprisingly complicated that it was proven to contain, at some point of its pattern growth process, any conceivable computation. Mathematicians call this a universal Turing machine or “Turing-complete.” This is not an intuitive concept. The information content of the underlying code is absurdly low, yet it can produce infinite complexity. What is more, there is no analytical method to tell you the pattern at iteration 1,000. If you want to know, you must play the game for 1,000 rounds, writing line by line. These systems are called cellular automata and are a beloved model for a branch of mathematics and the research field of Artificial Life (ALife). Some ALifers consider AI a subfield. Many AI researcher don’t care much about ALife. And neither of them care much about developmental neurobiology.

In information theory, the cellular automaton described above highlights an important alternative to describing complete endpoint information. Instead of a precise description of every detail of the pattern after 1,000 iterations, a complete description of the system is also possible by providing the few simple rules plus the instruction “apply these rules 1,000 times.” The information required to generate the complete system is also known as Kolmogorov complexity in algorithmic information theory. Data compression algorithms do exactly that. An image of a uniformly blue sky is easily compressed, because its algorithmic information content is low (paint the next 10,000 pixels blue). By contrast, a picture where every pixel has a random different color and no repeating patterns cannot easily be compressed. In the case of the cellular automaton, Kolmogorov complexity is very low, while endpoint information required to describe the system becomes infinite with infinite
iterations. The algorithmic information content required to create the system are a few instructions plus time and energy, while the endpoint information content is enormous in the case of many iterations.

The rule 110 cellular automaton provides us with a simple example of an algorithmic growth process that can generate more information based on simple rules, and yet its output can only be determined by letting it grow. “More” information is defined here as the information needed to describe the output if there were no growth process. However, in contrast to biological systems, rule 110 can only produce one fixed outcome with every iteration based on a set of rules that never change. For these reasons alone, rule 110 cannot be a sufficient model for biological systems. Yet, rule 110 teaches us that unpredictable unfolding of information is possible even with very simple rules in a deterministic system. For rule 110 there is a proof, the proof of Turing universality. For biological growth based on the genetic code, we face many more challenges: The rules are more complicated and change with every iteration of the running algorithm, and stochastic processes are central to its run. If a simple system like rule 110 can already be unpredictable, then we should not be surprised if algorithmic growth of biological systems turns out to be unpredictable. However, the proof for biological systems seems currently out of reach. The idea that information unfolding based on genomic information cannot be mathematically calculated, but instead requires algorithmic growth or a full simulation thereof, is a core hypothesis of this book.

A Shared Perspective

Biologists like to talk about the genes that contain a certain amount of information to develop the brain, including its connectivity. But in order to appreciate the information content of genes, we must understand the differences and consequences of information encoding for a self-assembling system versus a connectivity map. The genetic code contains algorithmic information to develop the brain, not information that describes the brain. It can be misleading to search for endpoint information in the genes or the mechanisms of the proteins they encode.
Address codes, navigational cues and key-and-lock mechanisms all follow such a rationale and make intuitive sense. And they all exist as molecular mechanisms, in brain wiring as elsewhere in biology. But they are part of unfolding algorithmic information, not endpoint information of brain connectivity. As the brain grows, different genes are turned on and off in a beautiful ballet in space and time, endowing each individual neuron with a myriad of properties that play out and change in communication with its neighbors. The neuron navigates as the city map grows and changes in interaction with the neuron’s own movement in it.

The study of genes in developmental neurobiology is a success story from at least two perspectives. First, in the quest for molecular mechanisms. What a gene product does at any point in time and space during brain development tells us something about a part of the growth program that is currently executed. But information about a specific molecular mechanism may only be a tiny part of the information that unfolds in the wake of a random mutation in the genome. A mutation can lead to more aggressive behavior of the animal. And yet, the mutation may well affect some metabolic enzyme that is expressed in every cell of the body. The molecular function of the gene product may tell us nothing about animal behavior. How the molecular mechanism of this gene is connected to the higher order behavior may only be understood in the context of the brain’s self-assembly, its algorithmic growth.

Many mutations have been found that change predispositions for behavioral traits, yet there may be only very few cases that we could reasonably call “a gene for a trait.” Most gene products contribute to develop the trait in the context of many other gene products, but do not contain information about the trait itself. A mutation, selected by evolution for behavioral changes, must change either brain development or function. If the effect is developmental, then we have to face the information problem: There may be no way to know what the altered code produces other than running the entire process in time (or simulating it on a computer). There may be no shortcut. This is the problem with the street navigation analogy: You have to navigate a changing map on a path that only works if the map changes just as you are navigating it.
The full route on the map never existed, neither at the beginning nor at the end of your trip, but instead the route was made in interaction with your own actions. This is the essence of self-assembly.

We can study self-assembly either as it happens in biology or by trying to make a self-assembling system from scratch. As of 2020, biological neural networks (i.e., brains) are still unparalleled in their intelligence. But AI is on it. And yet, self-assembly is not a major focus of AI. For many years, AI focused on formal symbol-processing logic, including enormous expert systems built on decision-making trees. As recently as the early 2000s, the victory of formal, logical symbol-processing AI was declared. Since then, just when some thought we were done with neural networks, a revolution has taken place in AI research. In the few years since 2012, practically every AI system used to predict what friends or products we allegedly want has been replaced with neural networks. “Deep learning” is the name of the game in AI today.

The ANNs we use as tools today are not grown by a genetic code to achieve their initial architecture. Instead, the initial network architecture is typically randomly connected and thus contains little or no information. Information is brought into an ANN by feeding it large amounts of data based on a few relatively simple learning rules. And yet, there is a parallel to algorithmic growth: The learning process is an iterative process that requires time and energy. Every new bit of data changes the network. And the order matters, as the output of a preceding learning step becomes the input of the next. Is this a self-assembly process? Do we ultimately need algorithmic growth or self-assembly to understand and create intelligence? One obvious problem with the question is that the definition of intelligence is unclear. But the possible role of self-assembly may need some explaining, too.

In the search for answers, I went to two highly respected conferences in late summer 2018, an Artificial Life conference themed “Beyond Artificial Intelligence” by the International Society for Artificial Life and the Cold Spring Harbor meeting “Molecular Mechanisms of Neuronal Connectivity.” I knew that these are two very different fields in many respects. However, my reasoning was that the artificial life and artificial intelligence communities are trying to figure out how to make something
that has an existing template in biological systems. Intelligent neural networks do exist; I have seen them grow under a microscope. Surely, it must be interesting to AI researchers to see what their neurobiology colleagues are currently figuring out—shouldn’t it help to learn from the existing thing? Surely, the neurobiologists should be equally interested in seeing what AI researchers have come up with, if just to see what parts of the self-assembly process their genes and molecules are functioning in.

Alas, there was no overlap in attendance or topics. The differences in culture, language and approaches are remarkable. The neurobiological conference was all about the mechanisms that explain bits of brains as we see them, snapshots of the precision of development. A top-down and reverse engineering approach to glimpse the rules of life. By contrast, the ALifers were happy to run simulations that create anything that looked lifelike: swarming behavior, a simple process resembling some aspect of cognition or a complicated representation in an evolved system. They pursue a bottom-up approach to investigate what kind of code can give rise to life. What would it take to learn from each other? Have developmental biologists really learned nothing to inform artificial neural network design? Have Alifers and AI researchers really found nothing to help biologists understand what they are looking at? I wanted to do an experiment in which we try to learn from each other; an experiment that, if good for nothing else, would at least help to understand what it is that we are happy to ignore.

So I assembled a seminar series, a workshop, about the common ground of both fields. The seminars are presented from the perspective of a neurobiologist who wants to know how our findings on brain development relate to the development of ANNs and the ultimate goal of artificial general intelligence. Many neurobiologists feel that ANNs are nothing like the biological template, and many AI scientists feel that their networks should not try to resemble biology more than they currently do. The seminars are therefore presented with a broad target audience in mind: there is so little common ground that it is easily shared with any basic science-educated layperson. The average neurobiologist is a layperson when it comes to AI, and most ANN developers are
laypeople when it comes to neurobiology. Developmental neurobiologists may feel they are not missing anything by not following the bottom-up approach of AI, and ANN developers may feel they are safe to ignore biological detail. But to decide what is not needed, it helps to at least know what it is we are choosing to not know.

One of the best outcomes of good seminars are good discussions. And here I didn’t need to search long. Going to conferences with these ideas in mind has provided me for years with experiences for how and where such discussions can go. I started writing this book with these discussions in mind. Initially, I only used them as a guide to pertinent questions and to identify problems worth discussing. As I kept on going back to my own discussions and tried to distill their meaning in writing, it turned out all too easy to lose their natural flow of logic and the associations that come with different perspectives. So I decided to present the discussions themselves. And as any discussion is only as good as the discussants, I invented four entirely fictional scientists to do all the hard work and present all the difficult problems in ten dialogs. The participants are a developmental geneticist, a neuroscientist, a robotics engineer and an AI researcher. I think they are all equally smart, and I do hope you’ll like them all equally well.

The Ten Seminars

The seminars of the series build on each other, step by step. Preceding each seminar is a discussion of the four scientists who exchange questions and viewpoints in anticipation of the next seminar. The series starts with The Historical Seminar: The Deeply Engrained Worship of Tidy-Looking Dichotomies, a rather unusual seminar on the history of the field. The “field” being really two fields, developmental neurobiology and AI research, this seminar provides an unusual and selective historical perspective. Their shared history puts each other’s individual stories in the spotlight of shared questions and troubles. Both struggle with remarkably similar tension fields between seemingly opposing approaches and perceptions. There are those who feel that the approaches, hypotheses and analyses must be rigorously defined for any
outcome to be meaningful. Then there are those who feel that, like evolution, random manipulations are okay as long as one can select the ones that work—even if that means giving up some control over hypotheses, techniques or analyses.

Both fields begin their shared history by independently asking similar questions about information. The discovery of individual nerve cells itself was a subject of divisive contention. Even before scientists were sure that separable neurons exist, concerns were already raised about the information necessary to put them all together in a meaningful network. Much easier to envision the network as a randomly preconnected entity. And when early AI researchers built their very first networks with a random architecture, they did so because they felt it had to be like that in nature—where should the information have come from to specifically connect all neurons? A randomly connected network contains little or no information; the network has to grow smart through learning. In biology, the dominance of this view was challenged already in the 1940s by studies that focused on the precision and rigidity of connectivity that is not learned. This work marked a turning point that led neurobiologists to ask questions about how network information can develop based on genetic information. By contrast, today’s artificial neural networks used in typical AI applications still only grow smart by learning; there is no genetic information. Yet, years in both fields played out in similar tension fields between precision and flexibility, between rigidity and plasticity. The fields may not have talked much to each other, but they mirrored each other’s troubles.

The historical background forms the basis for three sessions. The first session explores the types of information that underlie biological and artificial neural networks. The second session builds on the information-theoretical basis to discuss the approaches taken by biologists to understand how genetic information leads to network information—the missing element in most ANNs. The third session connects algorithmic growth to learning and its relevance for AI.

Each session consists of three seminars. The first session starts with Seminar 2: From Algorithmic Growth to Endpoint Information, which deals with the difference between information required to make
a system and information required to describe a system. Genes contain information to develop neuronal connectivity in brains; they don’t contain information that describes neuronal connectivity in brains. We are facing one of the hardest problems right from the start, mostly because human intelligence lacks intuition for this kind of information. The core concept is algorithmic growth. A set of simple rules is sufficient to create mindboggling complexity. But what is complexity? The journey to understand information encoding is intricately linked to this question. If a cellular automaton based on a very simple rule set can produce a Turing-complete system, including unlimited complexity of patterns, where is the information coming from? The algorithmic information content of the rules and the game is sufficient to create the entire system. This is very little information, and there is clearly no complexity there. On the other hand, the analysis of the pattern created by such a cellular automaton reveals unlimited depth. To describe the pattern requires a lot of information, something we like to call complex. All the while, the cellular automaton is a deterministic system, meaning repeated runs with the same rules will always produce the same pattern. The information for the development of this precision is somehow in the rules, but only unfolds to our eyes if the rules are applied iteratively, step by step, in a time- and energy-consuming process. This is the idea of algorithmic growth. The brain develops through algorithmic growth. Yet, in contrast to the cellular automaton, brain development includes nondeterministic processes and the rules change during growth. How useful is the analogy of the cellular automaton in light of these constraints? This question brings us back to the information that is encoded by the genetic code. When we discuss genes, we focus on biological neural networks. In the process, we learn about the type of information and the consequences of growth and self-assembly that define the network’s properties. These are the types of information that are typically left out in ANN design, and they may thus serve as a survey of what exactly is cut short in AI and why.

**Seminar 3: From Randomness to Precision** explores what happens when we add noise to algorithmic growth. Both an elementary set of rules for a one-dimensional cellular automaton or a genetic code will
deterministically produce identical results with every run in a precise computer simulation. But nature is not a precise computer simulation, or at least so we think. (Yes, the universe could be a big deterministic cellular automaton, but let’s not go there for now.) Biology is famously noisy. Noise can be annoying, and biological systems may often try to avoid it. But noise is also what creates a pool of variation for evolution to select from. From bacteria recognizing and moving towards sugar to the immune system recognizing and battling alien invaders, nature is full with beautifully robust systems that only work based on fundamental random processes that create a basis for selection. We will have some explaining to do, as we transition from the idea of simple rules that yet produce unpredictably complex outcomes on one hand to perfectly random behavior of individual components that yet produce completely predictable behavior on the other hand. Intuition may be of limited help here.

Awe and excitement about brain wiring mostly focuses on the exquisite synaptic specificity of neural circuitry that ensures function. As far as specific connectivity is absolutely required for precise circuit function, synaptic specificity has to be rigid. On the other hand, the brain develops with equally awe-inspiring plasticity and robustness based on variable neuronal choices and connections. In particular, neurons that find themselves in unexpected surroundings, be it through injury or a developmental inaccuracy or perturbation, will make unspecific synapses with the wrong partners. In fact, neurons are so driven to make synapses that scientists have yet to find a mutation that would prevent them from doing so as long as they are able to grow axons and dendrites and contact each other. Neurons really want to make synapses. If the right partner can’t be found, they’ll do it with a wrong partner. If a wrong partner can’t be found, they’ll do it with themselves (so-called autapses). This is what I call the synaptic specificity paradox: How can synaptic specificity be sufficiently rigid and precise to ensure function, if individual neurons are happy to make unspecific synapses?

The answer is closely linked to algorithmic growth: promiscuous synapse formation can be permissible, or even required, depending on when and where it occurs as part of the algorithm. For example, many
neurons have the capacity to initially form too many synapses, which contain little information. Through subsequent steps of the growth algorithm, this pool of synapses will be pruned and refined, thereby increasing the information content in the network. Rules for the weakening or strengthening of synapses are a core functional principle of all neural networks, both biological and artificial. This reminds us of brain function, learning and memory. But remarkably, neuronal activity can be part of the growth algorithm, long before there is even an opportunity for meaningful environmental input or learning. I call this *post specification*, the specification of synapses late in the developmental algorithm, following initially more promiscuous synapse formation. By contrast, synaptic *pre specification* occurs when only certain neurons get to see each other in space and time during their period of synaptogenic competency, i.e., the time window when they can make synaptic connections. If the patterns of the running algorithm restrict the synaptic partners that get to see each other, the problem of identifying the partner is greatly facilitated. The more spatiotemporal positions pre-specify partnerships, the more promiscuous, random synapse formation is permissible.

Random processes therefore need not be an enemy of precision in neural networks. Instead, random processes are abundantly utilized during algorithmic growth of the brain, just as in so many other biological processes. But random developmental processes do not necessarily produce variability in the outcome; they can also lead to almost perfectly precise synaptic connectivity patterns. And random developmental processes give rise to two of the most astonishing properties of biological brain wiring: flexibility and robustness. Connections not only change with experience, but also rewire in response to injury and developmental perturbation. ANNs also have some of these properties. And yet, historically, both neurobiology and AI had a rather strained relationship with randomness. Even today, most neurobiologists and ANN developers will consider noise as something to avoid, rather than as a design principle for a network. An understanding of the roles of noise will bring us closer to appreciating how to make networks flexible and robust in addition to making them with precision.
Seminar 4: From Local Rules to Robustness brings us back to the essence of self-assembly: local interactions during algorithmic growth. In order for local interactions to flexibly react to changing environments, local agents must be able to make their own decisions, independent of, and unknowing of, the larger system they create. This is the concept of autonomous agents. If the individual players of a soccer game would not make their own decisions, the game would be boring. If the players would not follow a common set of rules, the game would fall apart. The local interactions, the players’ decisions and flexibility, make the game interesting (if this kind of game happens to be interesting to you) and robust. The outcome is unpredictable at the level of the individual game, but the average outcomes over seasons are remarkably predictable. Which of the two, the individual game or the average season, is more interesting is in the eye of the beholder. For biological systems the beholder is evolutionary selection. For example, whatever local molecular and cellular interactions lead to different fingerprints may leave the outcome unpredictable at the level of the individual thumb, but perfectly predictable and robust at the level of selectable functionality.

In neural networks, both development and function vitally depend on individual neurons behaving as autonomous agents. The growing tip of a neuron employs random exploration of its environment through filopodia, tiny fingerlike protrusions. The neuron must be allowed to individually and locally decide whether it likes something it senses on the left or on the right using these protrusions. Similarly, the ability to learn in both biological and artificial neural networks relies on individual neurons, and individual synapses, to adapt their function. The concept of autonomous agents has made repeated stage appearances in AI. In all cases, the actions of autonomous agents only make sense in the context of a process that develops in time. The agents’ decisions and collective actions set the stage for higher order organization that develops step by step. They are part of self-assembly in space and time. And this brings us, at the end of session 1, back to the question of types of information. It is possible to describe, in arbitrary detail, the precise angles of the movements of every soccer player or a neuron’s growing protrusions. However, at what level of detail a situation must be described
in order to understand a distinct step of the underlying growth algorithm is not an easy question to answer.

In the second session we approach this question by diving into the realities of players and rules during the self-assembly of the biological brain prior to learning. All three seminars in this session focus on those aspects of the neurobiological history and their outcomes that are critical from the perspective of information theory: When and where does the information get into the network? ANNs used in AI today do not encode much information prior to learning; they are engineered and switched on for training. The second session is therefore all about biology, but with the goal to understand what it is exactly that ANNs are leaving out.

A powerful way to study brain development is experimental perturbation through mutation of the genetic code. After all, evolution did it before: genetic changes that affect development result in changes to the brain. The evolutionary approach is based on trial and error and does not require a need to predict the outcome of a genetic change as long as it can be selected. Selection of heritable, meaningful changes are evolution’s way of reprogramming the brain. But what are these meaningful changes to the genome? Are there special genes for the brain and behavior, or could any mutation in the genome help to reprogram the brain through information unfolding during development?

The second session starts with Seminar 5: From Molecular Mechanisms to Evolutionary Programming, in which we will explore these questions by analyzing how mutations can reprogram animal behavior. We will approach the answer through a discussion of programming by evolution: If a mutation causes heritable, meaningful and selectable change, then evolution can use it to rewire and reprogram the network. For this to work, it is not necessary that the functional mechanism of the protein encoded by the mutated gene is in any intuitive or direct way related to connection specificity. Rather, the effect of a mutation has to be such that the developmental process, and the unfolding of information that comes with it, reproducibly change the network. In this way, a behavioral predisposition can certainly be caused by a single mutation, yet there need not be a single “gene for that behavior.”
The fact that single mutations in the genome can reprogram animal behavior is well established. Pioneering experiments with fruit flies have surprised and enlightened this field for more than 50 years. Examples include mutants that affect courtship behavior and the internal clock that predictively guides behavior through the daily cycle of day and night. Importantly, the way such mutations were (and still are) found is based on accelerated evolution in the lab. The first step is to dramatically increase the occurrence of random mutations without any prediction as to what this might cause. The second step is to let thousands of the randomly mutagenized animals develop. The third step is to take those flies that survived the high mutation rate and assay them for behavioral alterations. These forward genetic screens for behavioral mutants led to the successful identification of mutants with altered behavior; over the years, several genes harboring these mutations were discovered. Some of the best studied of these genes are those where a molecular function directly relates to the behavior. There are beautiful examples, but they may be the exceptions. Most mutations that modify animal behavior affect genes that function in surprising ways during developmental growth, often at many steps or in many different cells. Such mutations can lead to heritable, meaningful and selectable behavioral change, but not through specific molecular mechanisms that are related to the behavior itself. Mutations may cause unpredictable developmental alterations that nonetheless lead to reproducibly different brains based on changes in network development or function. Those are the mutations that served evolution in the slow, trial-and-error reprogramming of brains and their astonishing behavioral innate programs. There is no single gene solely responsible for the spider’s ability to weave a species-specific web or the monarch butterfly’s ability to migrate a route of thousands of miles over a succession of generations. If our goal is to understand the programming of a neural network that accomplishes such feats, we must step beyond the idea of a single gene coding for a single behavior. We must learn how evolution reprograms the abilities of networks, including human intelligence.

**Seminar 6: From Chemoaffinity to the Virtues of Permissiveness**
deals with the historical and ongoing quest of developmental
neurobiologists to understand underlying molecular mechanisms. The invariable hope is to draw direct lines from mutations to genes to the gene products’ molecular mechanisms in trying to decipher neural network design. Neurobiologists prefer to characterize those genes whose gene products execute molecular mechanisms that make intuitive sense with respect to neuronal connectivity, hence the terms “guidance molecules” or “chemoattractants.” This is such a powerful idea and prominent concept in developmental neuroscience that we need to discuss examples of such molecules and their roles during algorithmic growth in some detail.

In search of information encoding for brain wiring, the holy grail has been the search for mechanisms that instruct the neuron where to make a connection. The idea of instructive mechanisms contrast with permissive mechanisms, which may be necessary to allow growth, but do not guide it actively. Oddly, the most widely used attractant for the guidance of growing neuronal protrusions in experimental culture is nerve growth factor—NGF. This is a molecule that the neuron needs to grow. By providing NGF only on the left, but not on the right, we can make neurons grow robustly to the left. This is clearly instructive. But wait, it’s a growth factor! The neuron simply will not grow where it is not present. That’s rather permissive. Obviously, a permissive mechanism (like a growth factor) can contribute to the neuron’s choice where to grow. From an information-theoretical perspective, the information for the directionality must have previously been provided in the location of the growth factor, which may lay out an entire path. The factor itself may be permissive, but the path it marks is instructive. Which brings us to the information needed to mark the path—and that brings us back to algorithmic growth where, step by step, paths may be laid out through the interactions of many autonomous agents, including the growing neuron itself. The path may not exist either at the beginning or the end of the neuron’s journey, but results from the interactions of the neuron with its surroundings as it grows. Some molecules on neuronal and non-neuronal surfaces convey local and temporally restricted attractive or repulsive signals. Yet other molecular mechanisms alter the propensity of the neuron to further grow extensions at all or gain or lose the
capacity to make synapses, or alter its mechanical interactions with the surroundings. In the context of algorithmic growth, the composite of all these factors determines the rules for each step in the algorithm. A genetic instruction need not be attached to a molecular mechanism of a single gene product. Instead, composite instructions are fleeting states of the system defined by the molecular and cellular contexts that happen to emerge at any given time and place during algorithmic growth.

Seminar 7: From Genes to Cells to Circuits is all about levels, from molecules to neural circuits, as we move towards the function of neurons in the network. How is it that in the field today the study of neural circuit function is obviously a question to be studied at the level of cells, while the study of the same neural circuit’s assembly is obviously a question to be studied at the level of molecules? This brings us back to the type of information encoded in the genome and its relation to processes at other levels. Single genes usually do not describe processes at higher levels, even though a specific mutation in a single gene can heritably and meaningfully change that process.

The levels problem is particularly pertinent when we are trying to span all the levels from the immediate effects of a mutation in the genome to a behavioral trait. Genome-wide association studies try to establish probabilities for a given genomic variation to be associated with a specific behavior. The probabilistic nature of the data and the difficulty to establish causality in such experiments is directly linked to the nature of algorithmically unfolding information.

Neuronal function is the level at which grown biological networks and engineered artificial networks meet. But in the biological template neuronal activity can in fact be part of the genetically encoded growth algorithm. Neuronal activity is part of information unfolding. As we have already discussed in the context of synaptic post-specification, activity is known to kick in before there is any environmental input. Correlated neuronal activity is one of the ingredients of algorithmic growth that require a random process to function robustly. It also provides a bridge to neural network function and AI.

The third session is all about transitions. First, there is the transition from development to function in neural networks. Next, the transition
from the naïve to the trained and functional network, and with it the transition from biological to artificial networks. The transition from dull to intelligent. And then there is the transition from separate biological and artificial networks to their interactive future. In all cases, the idea of information unfolding in a time- and energy-consuming manner serves as framework to assess possibilities and limitations.

In Seminar 8: From Development to Function we explore in what ways developmental self-assembly is relevant for network function. We will start with the burden of evolution and development for biological brains. Evolution had to work with the outputs of previous states, no matter how inappropriate they may have been when selection pressures changed. Evolutionary change happens in steps, however small, in a necessarily sequential manner. And the process takes, of course, time and energy. As a result, brains feature some remarkable and apparently nonsensical oddities that only make sense in light of development—and the way development was slowly shaped over millennia of evolutionary modification.

These kinds of biological oddities, and messiness, led computer enthusiasts who were trying to develop AI in the '80s to take pride in ignoring what their neuroscience colleagues were doing. “We can engineer things much better,” they may have thought, so why learn about the nonsensical solutions biology had to put up with?

And yet, if we avoid the burden of developmental history by starting with a randomly connected network prior to learning, the first problem we are confronted with is the time and energy it takes to train. And training, again, is a stepwise, time- and energy-consuming process. The order of input matters. And the ultimate function of the network is made possible, and burdened, by its training history. We will explore how information is stored in biological and artificial networks. How does the neural network save a four-digit PIN? The amount of bits needed to store this information is clearly defined in computational terms. Yet, neural networks save and retrieve this information flexibly and robustly, even if random neurons in the network drop out. In addition, the biological network does not have a separate training and function period. Learning is inseparable from using the network; storing is
inseparable from retrieving information. And again, we meet an evolutionary principle and the power of sequences in time. Many bits of information in the biological network—memories—can only be accessed by going through a sequence in time. Try saying your phone number in reverse order. How is this information content programmed, stored and retrieved?

If self-assembly is any guide, then information has to enter by changing the sequential, auto-associative network, which means it changes algorithmic information. Maybe memories should not be understood as stored entities at all, but rather as algorithmic rules sufficient to recreate them with a certain robustness, flexibility and variability. This bring us back to the cellular automaton that does not store the memory of the pattern at iteration 1,000, but instead the information to recreate this state. We will explore to what extent this process resembles algorithmic growth, and how it transitions from development to function.

In Seminar 9: From Algorithmic Growth to Artificial Intelligence we focus on artificial neural networks and their relationship to self-organization and algorithmic growth. We will finally also discuss definitions of self-assembly and intelligence. Most ANNs are based on the idea of an engineered design, flipping the on switch and training the network. By contrast, in biological networks the information encoding goes hand in hand with the development of the brain. The brains of a newborn, a toddler or a 10-year-old are clearly recognizable for their developmental stages morphologically, functionally and by their behavioral output. The question is whether a tedious, years-long process of self-assembly is a desirable step to create an artificially intelligent system. More specifically, is there ever a need to grow a neural network, or is training a predesigned network like in deep learning sufficient, maybe equivalent, or even superior?

A common idea in ANN development is that the product of development is only hardware infrastructure. A lot of biological idiosyncrasies can be congealed in a single parameter, like the synaptic weight. These are shortcuts that have served ANNs well for many years and many tasks. Yet, a key question associated with this reduction is how it may limit learning. In biology, the single parameter contains none of the
parameter space necessary for evolutionary programming to modify an algorithmically growing network. Based on these considerations, we dive deeper into the way engineered ANNs do, and do not, function.

Finally, in Seminar 10: From Cognitive Bias to Whole Brain Emulation, we will discuss the consequences of algorithmic information storage in neural network for the function and interaction of biological and artificial networks. We will start with a discussion of heuristics, the probabilistic nature of any information in the network. Neural network function is less the computing of input in conjunction with stored data based on logical operations, and more a process of probabilistic alignment and selection of patterns based on previous experiences. Both biological and artificial networks are biased by their experience. An ANN that has only been trained with numbers 0 to 9 will interpret the picture of an elephant as a number from 0 to 9.

We are all well-trained neural networks, but our brains come with a history track, as do ANNs. New information is not stored independent of other safely stored information content. Instead, any new bit of information is processed in the context of the entire history of the network. The same experience means something different for every individual. And the better the information is aligned with previous experiences, the easier it is for the network to “believe” the new arrival. This simple thought has some interesting consequences for the concept of cognitive bias: in a network built on algorithmic growth, bias is a feature, not a bug of the system, whether we like it or not.

Finally, if information is stored as an algorithmic process that requires time and energy, can it be retrieved and transferred in toto? That is, what does the self-assembling brain teach us about the potential to upload or download our brains? If information is not stored in any dedicated bits, but as algorithmic rules sufficient to recreate that information, then bandwidth of connection may not be the biggest challenge for data transfer. In the development of AI, we continue the debate about how similar artificial systems should be to the biological analog. But if we want to extend or copy our own brains, a clearer understanding of how information is actually stored or retrieved is needed. We encounter the levels problem again. To generate artificial human
intelligence, what parts of the algorithmic growth of the human brain can be cut short? In the design of ANNs, shortcuts are useful to shorten computation time by throwing out irrelevant detail. This approach works, as long as we do not need or want to simulate, say, spatially and temporally restricted modulation of many synapses through diffusible neuromodulators. But if we want to simulate human intelligence, don’t we need the neuromodulator, since circuit function requires synaptic changes that depend on the neuromodulatory context? We come to the question of the AI we want to generate. The shortcuts we choose in the development of artificially intelligent systems define what intelligence we get.

On Common Ground

This book was written with readers in mind that are interested in developmental biology or AI alike. However, those deeply immersed in either field will find much that is treated too superficially or from an unfamiliar perspective. I did not attempt to provide objective overviews over either field’s history or main achievements; many great works already exist on both accounts and are referenced throughout the seminars. My goal was to identify common ground, with a focus on underlying questions of information encoding. My hope is that a reader with deeper knowledge in either topic will still find reason to smile when trying to think what it may read like for someone in the other field.

I am convinced that all concepts presented here have been part of many ideas in different contexts before. Algorithmic growth in particular is not a new concept. It is implicit in all developmental processes and any attempt to understand how the genome encodes growing things. Yet, intuitive and mechanistic thinking in either field rarely considers the implications of unpredictable information unfolding. Those familiar with self-organizing systems may find most concepts presented here oversimplified, or my definition of self-assembly (seminar 9) wanting. Similarly, developmental neurobiologists are likely to find much that could have been added from the boundless list and beauty of molecular mechanisms underlying neural network development and function. But
common ground lies more in the motivation, the desire to understand how neural networks grow smart, than in the details of the individual disciplines. On this account, I hope the historical perspectives presented throughout the seminars may provide helpful parallels.

I am aware that many ALife and AI researchers may feel that reading a book written by a neurobiologist is not likely to be helpful for their work, both for reasons of perspective and the inevitable focus on unhelpful biological “messiness.” Similarly, some developmental neurobiologists may currently read a book or two on the application of deep learning to analyze their data, rather than to learn from the AI community about how “real” brains come to be. I started this project wishing there were a book that both would enjoy having a look at, or at least get sufficiently upset about to trigger discussion between the fields.
INDEX

Page numbers in italics refer to figures.

ablation: of cells or tissues, 70, 188, 239; of target, 68–70, 188
actin cytoskeleton, 183–85
action potential, 225
activity-dependent refinement, 119, 129–30
activity-dependent plasticity, 57, 119
Adami, Chris, 227
Adams, Douglas, 35, 86, 117, 141, 163, 190, 216, 242, 265, 289, 293, 314
adaptation: evolutionary, 249–51, 260; neuronal, 40–43
address code: and ID tags, 115, 196, 206; and relative positioning, 191; rigidity of, 70, 194–97; synaptic, 70, 188, 194–97, 206
adhesion molecule, 117, 127–28, 134. See also cell surface receptor
Administrative Behavior (book title), 294.

See also Simon, Herbert
AGI. See artificial general intelligence
aggression, gene for, 174, 226
aging, 274
AI. See artificial intelligence
alchemy problem, 78, 310
alcoholism, gene for, 226
alcohol sensitivity, 185
algorithm, definition of, 6
algorithmic function, 185, 257
algorithmic growth, 6, 92. See also developmental growth
algorithmic information, 6, 50–51, 98, 180, 311; and brain-machine-interfaces, 310–11; in contrast to endpoint information, 7–8, 98; and memory storage, 256; and synaptic specification, 212
algorithmic information theory, 6, 50–54, 87. See also algorithmic information
algorithmic probability, 54. See also Solomonoff, Ray
Alife. See artificial life
AlphaZero, 313
ANN. See artificial neural network
alternative splicing. See splicing
Amara’s law, 35
analogical tradition, 60
apple tree: and algorithmic growth, 87–88, 114–17, 123; genetic encoding of, 86–87, 101, 123; and variability, 101, 114, 144
Arbib, Michael, 55
artificial general intelligence, 10, 264–67, 278, 287; and ANN architecture, 79, 288; compared to biological intelligence, 27–32, 264–67, 278; and developmental growth, 305–7
artificial human intelligence, 27–28, 267, 278, 284, 306. See also human intelligence
artificial intelligence: and brain development, 304–5; definition of, 120, 278; evolution of, 277; history of, 47, 61–75; human-level, 76, 267; general (see artificial general intelligence)
artificial life, 6, 63, 92, 153, 277
artificial neural network: in AI applications, 12, 57, 236; comparison to biological neural network, 30–34, 48, 72, 83, 242, 281, 305–6; training of, 48–51, 243–44, 271–75, 299; first, 52 (see also Perceptron; SNARC); growing of, 146, 265
artificial retina, 290
artificial worm intelligence, 266, 306
Ashby, W. Ross, 156, 157
association study. See genome-wide association studies
attractive signal, 90, 120, 194, 204. See also guidance cue
attractor: in self-organizing systems, 268; molecular long range, 234
Austin, J. L., 77
autapses, 14, 118, 127
autism, 138
auto-associative memory, 256
autonomous agent, 16, 140–44, 148–59
autopoiesis, 163
Avery, John Scales, 50
axon: branches of, 150–52; choices of (see axon guidance); innervation of, 40–42, 68–71, 145, 152, 193, 196, 204
axonal competition, 194
axonal growth, 108, 126, 204, 220–23
axonal growth cone. See growth cone
axon-axon interaction, 132, 194
axon guidance, 41, 107–8, 158, 204–9, 220–23. See also chemoaffinity theory; debris guidance
axon pathfinding, 105, 158, 188, 193–96, 204–9, 220–23. See also axon targeting
axon patterning, 126–27, 132, 151–52. See also axon-axon interaction
axon regeneration, 46, 68–70
axon sorting. See axon patterning
axon targeting, 148, 205, 223
backpropagation, 27, 260, 271, 281–84
bacteria, 120
bacterial motor protein complex, 120
Baker, Bruce, 178
bandwidth, 290, 307
bee, 298
behavioral mutant, 18, 175, 177
behavioral trait, 8, 20, 174, 226
Bellen, Hugo, 183
Bennett, Charles, 51, 98
Benzer, Seymour, 167, 172, 173, 200, 201
Benzer paradox, 176
biased random walk, 121–22. See also chemotaxis
big bang, 112, 181
big data, 281
biological neural network: growth of, 45, 59, 68, 85; in a brain, 9, 13, 68; learning in, 27 (see also learning); synthesis with ANN, 304–5
bionic eye, 290
binary digit, 21, 49
bit. See binary digit
blind spot, 249
Blind Watchmaker, The (book title), 245
blueprint: in engineering, 4, 88–89 (see also endpoint information); and the genetic code, 5, 89
BMI. See brain-machine interface
Boltzmann’s constant, 99
Bonhoeffer, Friedrich, 58, 70, 193
Bostrom, Nick, 304
bottom-up approach, 10, 267. See also bottom-up technology
bottom-up technology, 142, 291. See also bottom-up approach
brain determinacy, 136; 4, 79, 119 (see also development; developmental growth; neuronal development); analogy to rule, 110, 187
brain development: and artificial intelligence, 304–5; continuity with brain function, 273; random processes in, 134–36; and self-organization, 278
brain upload: 304–11

For general queries, contact webmaster@press.princeton.edu
BrainGate, 308. See also brain-machine interface
brain-machine interface, 307–9
Braitenberg, Valentino, 130, 133
branching pattern, 114, 144, 152
Brenner, Sydney, 173, 199–200, 201–204, 217
burden of development, 21, 243
bureaucracy, 294
Butler, Samantha, 206

Caenorhabditis elegans, 128, 200–203
Cajal, Ramon y, 37–42, 100
calcium sensor, 184–85
capsules (in ANN development), 284
Carnap, Rudolf, 47
Carnot, Nicolas, 99
cartridge (in the fly brain), 251
CCD sensor, 246
cell adhesion molecule, 117, 127–134. See also cell surface receptor
cell autonomous program, of a developing neuron, 155–158, 195
cell division, 103
cell non-autonomous program, of a developing neuron, 155. See also cell autonomous program
cell surface receptor, 106–7, 114–19, 124, 150, 189–191, 212, 235, 254
cells that fire together, wire together, 48, 57–58, 119, 129–30, 273–74
cell types (differentiation of), 103–4, 274
cellular automaton, 5–7, 13–14, 94–97, 148
Chaitin, Gregory, 51, 98
chaos, deterministic. See deterministic chaos
Chedotal, Alain, 206
chemical tag, 172, 192–97. See also chemoaffine tag
chemical transmitter, 183–84
chemoaffine molecules, 46, 194. See also chemoaffinity theory
chemoaffine tag, 68–69, 106, 126. See also chemical tag; chemoaffine molecules; chemoaffinity theory
chemoaffinity. See chemoaffinity theory
chemoaffinity theory, 46, 68–73, 107, 130, 188–90, 195–99
chemoattractant, 19, 107, 205
chemoreceptor, 120–21
chemotaxis, 120–22, 241
chess, 51–53, 313
choices: human rational, 294; of a neuron during development, 34, 126, 140–146, 205, 213
Chomsky, Noam, 75
circadian clock, 177–178. See also daily rhythm
circadian rhythm. See circadian clock
circuit properties, 226
Clandinin, Thomas R., 132
Clausius, Rudolf, 99
code: algorithmic, 88–89, 109; genetic (see genetic code); molecular address (see address code); for a PIN, 29, 258, 266; for a target in brain wiring, 69–70 (see also target code)
cofactor, 178
cognitive bias, 77, 297–300
cognitive psychology, 278, 294–95
Cohen, Stanley, 107
columns, in the brain, 253. See also cortical column
commissural axon, 205
common sense, 296
compensation, in neural networks, 218–19
competition: axonal, 194; in self-organization, 271
complexity, 13, 87, 95–98; irreducible, 245; Kolmogorov (see Kolmogorov complexity); of neural networks, 197–98; unpredictable, 181–82
composite instruction, 20, 107–9, 117, 209, 214–16, 229
composite property, 209–10. See also composite instruction
compression (data): data, 6, 50–51; neural map, 68–70, 188
computation: based on von Neumann architecture, 255; in neural network, 255, 284; Turing's theory of, 53
computational evolution, 84, 277–78, 312
computational model, 72, 198
computational neuroscience, 55
computation time, 51, 98
connectionist view, 60–61, 192, 218, 279–80
connectome, 217. See also connectomics
connectomics, 200, 252. See also connectome
contrast enhancement (in vision), 246
convergent evolution, 250
convolutional network, 269–270
Conway, John, 86, 92–93, 141
Cook, Matthew, 94
coevolution (in self-organization), 271
cordyceps, 171
correlated neuronal activity, 20, 129
cortex, 253–57, 259, 270, 280–82, 290, 307–10
cortical column, 253–54, 289
courtship behavior, 178
Cowan, Jack D., 55
Crick, Francis, 173, 199
crystallins, 250
curiosity, 298
cybernetics, 49, 63, 130–32, 156, 164, 268–70
cytoskeleton, 185
daily rhythm, 176. See also circadian clock
Dartmouth workshop, 51, 61–62, 97, 294–96
data compression, 6, 50–51
Dawkins, Richard, 89, 140, 245
de Bivort, Benjamin, 135–38
debri guidance (of neuronal axons), 70
decentralized information storage, 29, 257–61, 310
decision-making process, 207, 294–97
deductive logic, 300
deep learning, 270, 280–81. See also machine learning
DeepMind, 313
default behavior (of neurons), 158–59
Delbrück, Max, 173, 224
dendritic self-avoidance, 123–26
dendritic tree, 100–101, 116–17, 123–26, 144, 152
dependency on context. See context
deterministic brain. See brain determinacy
deterministic chaos, 85–86
deterministic system, 13, 85–86, 95
development: burden of, 21, 243; precision of, 34, 131–32; stochastic, 54, 136–38
developmental constraint, 250–52
developmental growth, 178, 212–14, 254–61, 273, 280. See also algorithmic growth
developmental history, 21, 243–45, 249–254, 297. See also evolutionary history
developmental neurobiology, 8, 195
developmental robotics, 299, 313. See also robot
developmental timing, 70, 188
dichotomy: formal logic versus probability, 54, 61; nature versus nature (see nature versus nurture dichotomy); neat versus scruffy (see scruffy versus neat dichotomy); necessity versus sufficiency, 213; precision versus imprecision, 54; specificity versus plasticity, 46; tidy-looking, 54, 74, 77
Dickson, Barry, 178
differentiation (of cells). See cell types
disorder, neurodevelopmental. See neurodevelopmental disorder
dissipative system, 112
DNA-binding protein, 178. See also transcription factor
Dougherty, Ellsworth, 199
Drexler, Eric, 142, 302
Dreyfus, Hubert, 74, 78
Drosophila, 124, 171–74, 182, 217, 233
Dscam (gene name), 115–17, 127, 134, 150, 224–25
dynamic normality, 155, 163

E. coli, 120

Edelman, Gerald, 139, 211, 260–61

Edmonds, Dean, 52–53. See also SNARC

educated guess, 296. See also heuristics
electrical wiring, 4, 88
electrode, 290, 307–9
electrophysiological recording, 69, 73, 183
embodied entity, 276
embryonic development, 45, 197, 246–47, 260
emergent phenomena, 268
emergent property, 32
endpoint information, 5–7, 89, 98–101, 106–9, 252
energy landscape, 268
engineering: of ANNs, 273, 279–82; bottom up, 169, 187, 291, 302 (see also nanotechnology); electrical, 202; evolutionary approach, 170; genome, 185; interface with biology, 63–64, 310–11 (see also brain-machine interface); of proteins, 170; perspective versus biology, 245, 255, 279–82; reverse approach, 10; versus self-organization, 267, 279–82
entropy, 49–54, 99, 112, 273, 313–14
environmental information: contribution to brain wiring, 4, 36, 85–88, 119; as part of the genetic program, 15, 36, 59, 108, 119, 137; and the resonant theory, 41–42 (see also Resonance Theory); and spontaneous activity, 129–30 (see also activity-dependent fine-tuning); through learning (see learning); versus genetic information, 36, 48, 54, 85–88, 136, 137–38 (see also nature versus nurture dichotomy)
Eph receptors (gene name), 194–96
ephrins (gene name), 194–96
 equilibrium, 112, 268
equipotentiality (Karl Lashley), 44, 60, 253
ethanol sensitivity, 185
evolution, 137, 171, 247–52 (see also evolutionary programming; evolutionary selection); of a behavioral trait, 231–34; predictability of, 151, 181–82, 186
evolutionary algorithms, 63
evolutionary arms race, 171
evolutionary history, burden of, 246–249, 255, 261
evolutionary principle, 123, 137, 143, 187, 254, 260–61
evolutionary programming, 171, 181–85, 260–61
evolutionary psychology, 296
evolutionary selection, 101, 151, 245. See also evolution; evolutionary principle
exaptation, 250–52
expectation failure, 298
experience (in neural networks), 23, 258, 297–301
expression pattern, genetic, 212
eye-brain mapping, 130–32. See also retinotectal projections
eyeless/Pax6 (gene name), 178–79, 234

face recognition, 259–60
factual explosion, 179–81
feedback: in ANNs, 268–75; in artificial life research, 63; between autonomous agents, 146, 155, 222, 270; and compensation, 218; in the context of cybernetics and Norbert Wiener, 63, 164, 268–75; in developmental patterning, 125, 222; in a feedback loop, 125, 131, 177; between the genome and its products, 5, 103, 230; and molecular synaptic specification, 192, 197–99, 208–9; and unpredictability of outcomes, 168, 255
feedback loop, 125, 177
feedforward neural network, 269–71
Feynman, Richard, 37
fingerprint, 136–37
flagellum, 120–22
Flanagan, John, 193
Fleming, Roland, 270, 280
flexibility: developmental, 68–70, 199, 202; of neural networks, 59–60, 257
floor plate, 204–5
fly room, 174, 224
formal logic, 9, 47, 54, 61–63, 295
forward genetic screen, 18, 166–68, 182, 203
Franklin, Rosalind, 173
Friedberg, Errol, 200
Friedberg, Richard, 63
fruitless (gene name), 178–82, 228, 234
fruity (gene name), 178
functional plasticity, 54, 57, 73
functionalist view, 45

game of life, 86, 92–94
gating, (neural circuit), 186
gay gene, 208–31
gene map, 174,
gene product: developmental function of, 8, 103, 169, 223, 274; feedback on genome of, 103, 274
general chemoaffinity, 196
generality, 180, 207
general principle, 180, 207
genetic algorithm, 63
genetic code, 5–7, 88–90, 108–9, 140; feedback on, 103; outcome predictability of, 5–7, 230; replication errors of, 137
genetic determinism, 84, 136–37
genetics, polygenic. See polygenic genetics
genetic sensitization. See sensitized background
genetic screen. See forward genetic screen
genetic sufficiency, 178, 213
genome: in ANN evolution, 277; and behavior, 172, 228 (see also genetic basis); as a blueprint, 89; and encoding of stochastic processes, 137; feedback with its products, 103, 274 (see also feedback; transcription factor cascade); information content of, 48, 109, 221–22; what is encoded by, 59, 72, 89, 106, 221 (see also gene product)
genome-wide association studies, 227–31
Gill, Kubir, 177
Golgi, Camillo, 37–41
Gould, Stephen Jay, 249–51
GPS (global positioning system), 4
gradient: from instructive to permissive, 209, 213–14; in neurodevelopment, 71, 189, 194–96
gradient descent, 270–71
grandmother neuron, 258
Grassé, Pierre-Paul, 156
growth cone, 106 (see also axon pathfinding; filopodia); and attraction of repulsion, 194, 204 (see also axon pathfinding); as autonomous agent, 139–41, 148–52, 154
growth factor: as part of composite instructions, 108, 206; as permissive or instructive signal, 19, 107, 158–59, 206 (see also nerve growth factor)
guidance cue, 90–91, 187–88; cell surface molecules as, 190–91; as permissive of instructive signal, 107, 234; roles during brain wiring, 190–91, 234, 305
guidance molecule, 88. See also guidance cue
guidance receptor. See guidance cue
gut feeling. See heuristics
GWAS (genome-wide association study).
See genome-wide association studies

Haeckel, Ernst, 246–47
half-te
ctum, in classic regeneration experiments, 68–70
Hall, Jeff, 176, 178
Hamburger, Viktor, 44
hardware infrastructure, and role for neural network implementations, 286
hard-wired neural network, 36, 133
Harrison, Ross, 38–39
Hassan, Bassem, 124
Hawkins, Jeff, 67–68, 255–56
Hebb, Donald O., 47–48, 52, 268
Hebbian learning rule, 48
Hebb’s law. See Hebbian learning rule
Hebb synapse, 48, 129
Heberlein, Ulrike, 185
heterosexuality, genetics of, 228–30. See also homosexuality
heuristics, 295–97
higher-order property, 157, 236
Hinton, Geoffrey, 267, 276, 283–84
Hintze, Arend, 277
Hitchhiker’s Guide to the Galaxy, 35, 163
Hodgkin and Huxley, model for action potential, 241
Holland, John, 63
homologous response, as defined by Paul Weiss, 40
homophilic repulsion, 225. See also Dscam homosexuality, gene for, 226, 230. See also homosexuality
honey bee, 298
Hope, Tony, 72
housekeeping genes, 166, 183
Hubel, David, 74, 253
human intelligence, 267, 275–76
identification tag. See molecular identification tag
image classification, 276, 286
immune system, comparison with nervous system function and development, 139, 260
immune system, function of, 114–16, 119, 123
imprecision: in science, 54; and selection, 114
incomplete information, in decision making, 295
information: algorithmic, 5–7, 49–50, 95–98; endpoint, 98–102, 157; and energy, 99–100; environmental, 129, 137; genetic, 103, 108–9, 137, 175, 226; incomplete (see incomplete information); irrelevant, 101–2, 207; missing (see information entropy) storage, 29–30, 257 (see also memory); unfolding, 95–98, 193, 179–81
information entropy, 50, 99
information problem, 1–3, 38, 68–71, 175, 197
information theory, 49–51
innate behavior, 45, 136
instructive mechanism, 19, 107, 117, 193, 208–10. See also composite instruction
instructive signal. See instructive mechanism
intelligence, 72, 153; artificial (see artificial intelligence); definition of, 278–79; gene for, 174–76, 226; human (see human intelligence)
intelligent machines, 304
interface, of brain and machine, 290, 307–10
invariant representation, 256–59
irreducible complexity, 245
iteration, 6–7, 96
Jacob, François, 173
Jessell, Thomas M., 204
judgement heuristic, 296–97
Kahneman, Daniel, 295
Keating, Michael, 198
Kirschfeld, Kuno, 130–33
Kirschner, Marc, 179
knock-out, 105, 150, 221, 233
Kolmogorov, Andrey, 51, 98, 268
Kolmogorov complexity, 6, 98
Konopka, Ron, 176–77
Kurzweil, Ray, 278
language processing, 270, 306
Laplace, Pierre-Simon, 304
Laplace’s demon, 304
Lashley, Karl S., 44–47, 59–60, 134. See also equipotentiality
lateral inhibition, 124–25, 272
learning: and algorithmic function, 243, 275; of ANNs, 45, 65, 273, 282; with backpropagation, 27, 281; of biological brains, 48, 137, 299; of brain-machine interface, 310; deep (see deep learning); as continued development, 30, 79, 119, 137, 265, 280–83; and memory, 48, 274; supervised, 27, 263, 269, 281; unsupervised, 30, 271
learning history, 264
learning mechanism, 48, 275
learning rules, 48. See also Hebb, Donald O.
Lee, Mark H., 299
Levi-Montalcini, Rita, 107
Lewis, Edward B., 174, 224
Lewontin, Richard C., 136, 250
Lighthill report, 74
light-sensing cell, 131, 246–49
limbic system, 307
Lindenmayer, Aristid, 87, 101–2
linguistic theory, 75
little King Kong, 170–71
local decision, 140–44, 155
local interaction, 148, 207–71
lock-and-key mechanism, 114, 127
logical depth, 98
Logic Theorist, 62, 294–95
long-range attraction, 207–8
L-system, 87, 101–12
machine evolution, 63
machine learning, 109. See also deep learning
mass action (Karl Lashley), 44
master regulator, 261, 272–73. See also eyeless/Pax6
matchmaking, synaptic, 117–18, 126–27, 196, 208
Maturana, Humberto, 163
Maxwell, James Clerk, 99
Maxwell’s demon, 99
McCarrick, John, 61–62, 76, 283
McCorduck, Pamela, 78
McCulloch, Warren S., 46–47, 295
McCulloch-Pitts neuron, 47, 53, 60–64, 305
mechanism: general, 207; instructive (see instructive mechanism); molecular (see molecular mechanism); permissive (see permissive mechanism)
medulla (fly brain structure), 221–23
membrane receptor, 179. See also cell surface receptor
memory, 242, 256–61 (see also learning); computer, 91; invariant, 256; retrieval, 256; storage, 28–30, 256
memory-prediction system, 255, 279–80, 298. See also Hawkins, Jeff
Menzel, Randolf, 298
metabolic enzyme, 103, 166–69, 190–91, 232
metabolism, role for brain wiring, 113, 183, 286
midline crossing, 205
migration, of monarch butterfly, 2, 31, 172, 230, 260
Miller, George, 52
mind-body problem, 295
minimum: global, 268, 271; local, 271
missing information, 49–50, 89, 99, 314
Mitchell, Kevin J., 136
molecular address code. See address code
molecular assembler, 291, 302–3
molecular clock, 167, 177
molecular code, 199, 214
molecular dynamics simulation, 303, 306
molecular identification tag, 72, 115–16, 206.
See also chemoaffinity theory; Sperry molecule
molecular machine, 142, 177, 302
molecular mechanism, 149–50, 195 (see also instructive mechanism); and the levels problem, 216, 219–20; relevance of for phenotypic outcome, 164–68, 177–79, 184, 192, 204
molecular nanotechnology. See nanotechnology
molecular target code. See target code
monarch butterfly migration, 2–3, 31, 172, 230–31
Monod, Jacques, 173
monozygotic twins, 136–38
Morgan, Thomas Hunt, 174, 182
morphological rigidity, 57, 76
motion detection, 264
motion vision. See motion detection
motor cortex, 290, 307
motor protein, in bacterial chemotaxis, 120
Mountcastle, Vernon, 253
multilayer perceptron, 269
Musk, Elon, 307–10
mutagen, to induce mutations for genetic screens, 18, 166, 175–78, 203
mutation, 8, 18, 105, 113–15, 167 (see also evolutionary programming; single nucleotide polymorphism); and predictability of phenotypic outcome, 85, 167–69, 174–76; random, 132 (see also mutagen)
mutation rate, 166, 203
MuZero, 313

nanomachine, 142–43
nanoscale, 302

nanotechnology, 142, 291, 302
Nash, John, 61
natural language, 74–75
nature versus nurture dichotomy, 42, 54, 136, 137
necessity and sufficiency (in genetics), 178, 213, 234
feedback
Nell2 (gene name), 206
nematode, 199. See also Caenorhabditis elegans
neocortex. See cortex
neocortex column. See cortical column
NGF. See nerve growth factor
nerve growth factor, 19, 107–8, 158–59, 206
netrin (gene name), 204–9, 212, 234, 305–6
network topology, 36, 252, 273, 280
neural circuit, 216, 217–19, 225
neural Darwinism, 139, 260
neural group selection, 139, 260
neural network: artificial (see artificial neural network); biological (see biological neural network); compensation in (see compensation); complexity of, (see complexity); experience in (see experience); feedforward (see feedforward neural network); flexibility of (see flexibility); hardware of (see hardware infrastructure); hardwired (see hard-wired neural network); random architecture of (see random architecture); random connectivity of (see random connectivity); recurrent (see recurrent neural network); redundancy in (see redundancy); relevance of growth for (see relevance); robustness of (see robustness); self-organization in (see self-organization)
nanomachine, 142–43
neurodevelopment. See neuronal development
neurodevelopmental disorder, 138

For general queries, contact webmaster@press.princeton.edu
neurogenetics, 128, 172, 217
neuromodulator, 24, 33, 240–41
neuron: artificial, 47, 60, 102, 284 (see also McCulloch-Pitts neuron); biological, 1, 12, 37, 100, 108, 274; in culture, 158
neuronal aging, 274–75
neuronal activity: 15, 59, 128–29, 309; genetically encoded, 57–59, 129, 137; under optogenetic control, 218–19; spontaneous, 129
neuronal circuit. See neural circuit
neuronal connectivity, 13, 195, 248
neuronal development, 104, 123–28, 145, 150–59, 195, 221–23, 274 (see also algorithmic growth; brain development; branching pattern; development); genetic contribution to, 4, 8, 108, 179; molecular mechanisms in, 104–5, 195
neuronal differentiation: 103, 146
neuronal excitability, 129, 226
neuronal identity, 115
neuronal group selection. See neural group selection
neuronal growth, 108, 123–28, 144, 150–59, 221–23. See also algorithmic growth; neuronal development
neuronal properties, 226, 236
neuron doctrine, 37–40
neuron types, 42, 197, 248. See also neuronal differentiation
neuropil (synaptic brain region), 221–23
neurotransmitter, 183, 305
Newell, Allen, 62, 283
noise: in ANNs, 271; as a pool of variation, 14–15, 115, 123–25, 129, 134; and randomness, 95, 114, 136
nonadditive genetic effect, 230
Notch signaling, 124–25, 272
Nüsslein-Volhard, Christiane, 179
one-to-one mapping, 71–72, 89, 196, 212, 227
ontogeny, 246
open system, in thermodynamics, 112
OpenWorm project, 203, 236, 241
optimization function, 271
optimization problem, 294
optogenetics, 218–21
order from noise, 156
output error, in ANNs, 271. See also backpropagation
Papert, Seymour, 65–67
parameterization, 225
parameter space: in ANN function, 271; for evolutionary programming, 23, 286
penetration: and genetic context, 229–34; phenotypic, 108, 114, 176, 213
Perceptron (first ANN), 63–66, 269, 272. See also Rosenblatt, Frank
Perceptrons (book title), 65, 281
period (gene name), 176–78, 187
period, developmental, 274
permissive mechanism, 19, 107, 117, 193, 206–10, 305
permissive signal. See permissive mechanism
phenotypic penetrance. See penetrance
phenotypic variability. See variability
pheromone, 155–56
phototaxis, 175, 183
phylogeny, 246
Pitts, Walter, 47–48. See also McCulloch-Pitts neuron
place cells, 284
plasticity: developmental, 12, 40–44, 54, 199; of network function, 57, 119, 274, 290
pluripotent tissue, 234
polygenic genetics, 229–30
pool of variation, 14, 114, 123, 137, 151, 170
positive feedback loop, 125
post-specification, synaptic, 15, 130
Poulson, Donald, 178
precision: based on noise, 114, 122–26, 129; of development, 10, 132, 302; versus flexibility, 34, 42; of network wiring, 88, 129, 132; as a secondary consequence, 252 (see also spandrel)
predisposition, genetic, 8, 176, 215–16, 227
pre-specification, synaptic, 15, 130–34
Principia Mathematica, 47, 53–54, 62, 295
probabilistic biasing, 130, 134,
probability theory, 49, 54
probability: algorithmic, 54; in biased
random walk, 121–22; of phenotypic
outcome, 85, 138, 176, 227–29 (see also
penetrance)
programmed cell death, 145, 272
prokaryote, 120
promiscuous synapse formation, 14–15,
126–28, 130–34
protein design, 187
protein interaction network, 226
proteome, 105
protocadherins (gene name), 225
protocognition, 120
pruning, of synapses of neuronal branches,
119, 130. See also synaptotropic growth
Purkinje cell, 100–101, 152
random architecture, of neural networks, 12,
118. See also random connectivity
random connectivity, of neural networks,
72, 118, 264, 292
random factors, in genetics, 136
random process: in activity-dependent fine
tuning (see activity-dependent refine-
ment); during algorithmic growth, 15,
57–59, 115, 134, 186–87; in branching, 123;
in evolution and selection, 14–15, 84, 123;
and flexibility, 82; indistinguishability
from unpredictable deterministic
growth process, 149; as the opposite of
precision, 15; and precise outcomes, 59,
128–29; randomness, definition of: 98;
and robustness, 20, 84, 128–29; and
variability, 57–59
rational choices, of humans, 294–301
receptive field, 270
receptor protein, 105. See also cell surface
receptor
recognition molecule, 188–91. See also
chemoaffinity theory
recurrent neural network, 77, 270–72, 281
recursive process, 260
reductionism, 110–11
reductionist’s dilemma, 223–24
redundancy: genetic, 208, 231–32; in
information storage, 29; in neural
networks, 218
regeneration experiments, 40, 46, 68, 188
reinforcement learning, 273, 313
relative positioning, during network
development, 71–73, 188–91, 194–97, 222
relevance: in evolutionary selection, 136; of
growth for neural networks, 45, 126; of
information (see relevant information)
relevant information, 108
representation: internal, 75, 297; invariant
(see invariant representation)
representativeness, 296. See also generality
repulsive signal, 71, 123–24, 150, 196, 204, 225
Resonance Theory, 40–42. See also Weiss,
Paul Alfred
reticular theory, 38, 83
retina, 129, 246–48
retinal activity waves, 59, 119, 129
retinal ganglion axon. See retinal ganglion
cells
retinal ganglion cells, 130, 193–94, 248
retinotectal projections, 69, 72, 127, 195.
See also eye-brain mapping
robos (gene name), 254
robot, 26–34, 141–42, 187, 312–13. See also
developmental robotics
robustness: of neural network develop-
ment, 15, 148–49, 154; of neural network
function, 55, 60, 257
roof plate, of spinal cord, 204–5
Rosbash, Michael, 176
Rosenblatt, Frank, 63–67. See also Perceptron
Rothenfluh, Adrian, 185
Rubin, Gerald M., 217
rule, definition of, 235
rule 110, 5–7, 94–97, 109–10
rule of thumb, 295. See also heuristics

Sanes, Joshua, 128
Schank, Roger, 74–75, 298–99
schizophrenia, 138
Schmucker, Dietmar, 134
Schrödinger, Erwin, 173, 224
scientific method, 300
scruffy versus neat dichotomy, 75–77, 130, 134, 170, 187
secondary rationalization, 249
second law of thermodynamics, 99
selection, evolutionary. See evolutionary selection
self-amplification, 271
self-assembling brain, 109, 181–82, 222–24, 282; and aging, 275
self-assembly, 9, 198, 222–24; and autonomous agents, 146, 148; definition in this book, 273; and simple rules, 110
self-avoidance: of branching neurons, 116–19, 123, 150, 225; gene for, 150, 225–26
self-modifying process, 155. See also
dynamic normality; feedback; transcription factor cascade
sensitized background, in genetics, 232–33
sensory cortex, 253, 307
sensory system, 120, 248
sequence: and algorithmic information, 98; in developmental growth, 275; in memory storage (see sequential memory)
sequential memory, 244, 256, 259–61, 275, 299
sexual behavior, 178, 227–28
Shannon, Claude, 49–53, 62, 99
Shannon entropy. See information entropy
Sharma, Sansar, 68–69, 71, 127, 130, 198
Shatz, Carla, 57–59, 128–29, 236
shortcut: to creating intelligent systems, 3, 236, 272, 275, 301–6; to predicting
outcomes of growth processes, 94–98, 106, 227
short-range attraction, 205
signaling molecule, 120, 125, 185, 272
Simon, Herbert, 62, 283, 294–95
simple rules, 110, 235
single cell sequencing, 104
single nucleotide polymorphism (SNP), 227
sliding scale model, 70, 191
Smalley, Richard, 302
Smith, Stephen, 152
SNARC, first ANN machine, 52–53, 197.
See also Minsky, Marvin
SNP. See single nucleotide polymorphism
soccer, 146–49
Solomonoff, Ray, 50–54, 62, 97, 242
somatosensory cortex, 253, 290, 307
sonic hedgehog (gene name), 206, 234
spandrel, 250–51
spatiotemporal specificity, 15, 206, 212
speech recognition, 283
Sperry, Roger Wolcott, 42–47, 54, 57, 68–77, 83, 174, 188–92, 193–98. See also chemoaffinity theory
Sperry molecule, 79, 106, 189. See also
chemoaffinity theory; guidance cue
spinal cord, 38–39, 90, 204–6
spinal motor neurons, 151
splicing: alternative, 134, 137; error rate, 137
spontaneous activity, 129. See also activity-dependent refinement
squirrel monkey, 253
states, neuronal during algorithmic growth, 274
stigmergy, 156
stimulation (neuronal), 128, 291
stochastic dynamics, 125, 137, 152, 306
stochastic process, 125, 134–37, 150–52, 229
strict chemoaffinity, 70, 196–98. See also
Sperry, Roger Wolcott
Sturtevant, Alfred, 174
Südhof, Thomas C., 183
supernumerary, 304, 312
supervised learning, 27, 263, 269, 281
surprise, and information, 256, 293–94, 298–301
swarm behavior, 140
symbol-processing logic, 3, 9, 63, 153
symmetry-breaking mechanism, 125
synaptic matchmaking, 117–18, 208
synaptic modulator. See neuromodulator
synaptic promiscuity, 126–28, 133
synaptic specificity, 14, 126–28, 202, 208
synaptic specificity paradox, 14
synaptic strength, 33, 242, 260
synaptic weight: in ANN, 22, 258, 266, 277, 284–86; in the brain, 110, 280
synaptogenic competency, 15. See also composite instruction
synaptotagmin (gene name), 183–84, 187
syntax, in language and linguistics, 47, 75
systems matching, 68–70, 73, 188, 191, 198
phenotypic, 15, 34, 134–36, 230, 252; pool of (see pool of variation)

Ulam, Stanislaw, 92
uncertainty: in decision making, 296; in information theory, 50 (see also undecidability; unpredictability)
uncoordinated (unc) mutants, 204
undecidability, 86, 94, 96
unfolding of information, 214, 244, 255.
generalized system, 6, 94–96. See also Turing-complete system
universe: and determinism, 14, 95, 182, 304; and increase of entropy, 112–13, 181, 313–14
unpredictability: of developmental outcome, 148, 170, 182; of information unfolding, 186, 255 (see also rule 110)
unreliability, of components in biological and artificial systems, 53, 55
unsupervised learning, 30, 271, 281

Varela, Francisco, 163
variability: in the context of nature vs nurture, 42, 54, 57, 136–37; of developmental outcomes (see phenotypic in this entry); of developmental processes, 124–25, 147; phenotypic, 15, 34, 134–36, 230, 252; pool of (see pool of variation)
Vaughn, James, 151–52. See also synaptotropic growth
visual cortex, 253, 270, 290
visual perception, 270
von der Malsburg, Christoph, 72–73, 109
von Foerster, Heinz, 156, 271
von Gerlach, Joseph, 38–39
von Neumann, John, 51, 55, 61, 91–92
von Neumann computer architecture, 51–53, 91, 255–58
von Waldeyer-Hartz, Gottfried Wilhelm, 37–38

Watson, James, 173, 199, 224
Weiner, Jonathan, 174
Weiss, Paul Alfred, 40–43
What Is Life? (book title), 173, 224
White, John, 128, 202
whole brain emulation, 304–6

Wiener, Norbert, 49–50, 63, 130, 268
Wiesel, Torsten, 74, 253
Wilkins, Maurice, 173
Willshaw, D. J., 72, 109
Winograd, Shmuel, 55
Wolfram, Steven, 94–95, 181–82, 186

XOR, perceptron limitation, 281

Young, Michael, 176

zebrafish, 254
zero sum game, 147
Zipursky, S. Lawrence, 132