

Article

Circuit Design in Biology and Machine Learning. II. Anomaly Detection

Steven A. Frank 

Department of Ecology and Evolutionary Biology, University of California, Irvine, CA 92697-2525, USA;
saf@stevefrank.org

Abstract

Anomaly detection is a well-established field in machine learning, identifying observations that deviate from typical patterns. The principles of anomaly detection could enhance our understanding of how biological systems recognize and respond to atypical environmental inputs. However, this approach has received limited attention in analyses of cellular and physiological circuits. This study builds on machine learning techniques—such as dimensionality reduction, boosted decision trees, and anomaly classification—to develop a conceptual framework for biological circuits. One problem is that machine learning circuits tend to be unrealistically large for use by cellular and physiological systems. I therefore focus on minimal circuits inspired by machine learning concepts, reduced to the cellular scale. Through illustrative models, I demonstrate that small circuits can provide useful classification of anomalies. The analysis also shows how principles from machine learning—such as temporal and atemporal anomaly detection, multivariate signal integration, and hierarchical decision-making cascades—can inform hypotheses about the design and evolution of cellular circuits. This interdisciplinary approach enhances our understanding of cellular circuits and highlights the universal nature of computational strategies across biological and artificial systems.

Keywords: biological design; evolution; artificial intelligence; boosted decision trees; dimensional reduction; internal model principle



Academic Editor: Pavel Kraikivski

Received: 15 July 2025

Revised: 14 August 2025

Accepted: 22 August 2025

Published: 24 August 2025

Citation: Frank, S.A. Circuit Design in Biology and Machine Learning. II. Anomaly Detection. *Entropy* **2025**, *27*, 896. <https://doi.org/10.3390/e27090896>

Copyright: © 2025 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Many biological circuits sense danger. Some respond to common molecular patterns associated with attack. Others perceive environmental threats for which fear or fighting may be helpful [1–5].

An unusual or surprising environment provides another clue of danger. For example, the absence of an expected event could signal an anomaly. The famous comment by Sherlock Holmes about the dog that did not bark illustrates an anomalous absence of an expected event [6].

A Scotland Yard detective asked Holmes: “Is there any other point to which you would wish to draw my attention?” Holmes answered: “To the curious incident of the dog in the night-time.” The detective replied: “The dog did nothing in the night-time.” Holmes countered: “That was the curious incident.”

Intuitively, humans have a sense of anomaly, when unexpected events trigger heightened alertness. The word “eerie” captures the notion of discomfort when “things don’t add up” in an unfamiliar situation.

For these reasons, anomaly detection focuses on deviations from what is typical. An anomaly detection circuit must learn an internal model of the typical pattern. Any departure from that model triggers a warning. This approach contrasts with detecting specific danger signals that directly indicate peril, instead emphasizing deviations from common observations.

In mammalian brains, hippocampal circuits detect anomalies [7–9]. Immune systems may have such circuits [10,11]. Self versus nonself recognition is not fully understood [12] and might, in some cases, depend on detecting anomalous patterns as nonself. Plants might use anomalous volatile organic compounds of neighbors as nonspecific danger signals [13,14]. However, few biological studies emphasize nonspecific anomaly detection.

This article introduces anomaly detection in machine learning [15–18]. Computational models use a wide variety of circuit types to detect anomalies. Those different types of computational circuits suggest the kinds of biological circuits that might detect anomalies. Because anomaly detection is a type of classification problem, aspects of this topic also provide insight into other biological classification challenges.

2. Contributions of This Work

2.1. Overview of the Series

This article continues the series on circuit design in biology and machine learning [19]. The series uses insights from machine learning to understand how evolutionary processes build biological circuits. The first article in the series introduced the motivation and challenges for linking biological and machine learning circuits, with examples [19]. This subsection adds further background.

Three facts suggest that machine learning may provide insight into the evolutionary aspects of biological design. First, machine learning and biological organisms often face similar challenges. How can environmental inputs be classified into categories? How can a system predict future inputs? What is the best response to a type of environment?

Second, natural selection is one type of learning algorithm. Machine learning deploys a broader range of algorithms. But those different algorithms tend to modify systems in broadly similar ways [20–22].

Third, machine learning and biology often solve challenges by using a computational network to build an input–output response circuit. Here, we think of a biochemical network as a kind of circuit that takes inputs and computes outputs. When machine learning computes solutions without an explicit network, usually the computation can be embedded within a network to achieve the same result.

The fact that machine learning and biology typically build responses by creating computational circuits means that we can study how machine learning solves particular kinds of problems and use those solutions to make predictions about how evolutionary processes design biological circuits to solve the same sorts of challenges.

This series emphasizes simple biochemical circuits, primarily in cells. The analogy between neurobiological and machine learning circuits is well known, although directly linking the architecture and function of biological and computational circuits remains an ongoing challenge [23–27].

By contrast, relatively little work has been conducted to match cellular or physiological circuits to common machine learning architectures. Two challenges arise.

First, although many biochemical circuits in cells have been identified and partially understood, it is not easy to describe complete circuits, understand their computational architecture, and evaluate the sorts of computations that are used to achieve their function.

Second, computational networks in machine learning tend to be much larger than could reasonably fit within a cell. Thus, we must develop new machine learning models that emphasize greatly reduced size.

Given those constraints, this series primarily aims to outline a new theory that links these two subjects. Some general predictions arise about the architecture of biological circuits. Overall, these articles show the broad conceptual links between particular external challenges and the types of biological circuits that may be favored by evolutionary processes.

2.2. Insights from Anomaly Detection

This article develops the following points, often with simple illustrative models and example quantitative analyses.

- Machine learning provides new ideas for how cellular and physiological circuits may solve anomaly detection.
- Some challenges require evaluating a single atemporal multivariate input for anomalies. Others require estimating deviations from recent temporal trends. Simple models illustrate different circuit designs for atemporal and temporal cases.
- Detecting anomalies often requires evaluating multivariate patterns in inputs by integrating signals from ensembles of sensors or receptors. This article reviews basic measures of signal information.
- Digital sensors reduce continuous analog inputs to discrete binary outputs, losing information but also reducing sensitivity to noise and measurement error. Digital sensors are easier to implement and easier to combine into broader circuits.
- Machine learning uses large circuits. Cells require small circuits. This article shows that small circuits can achieve significant resolving power.
- Some anomalies differ in mean input values. Summing the inferences by individual sensor outputs provides a good response.
- Other anomalies differ in correlations between inputs. Decision trees work well, each sensor responding within a sequence based on the output of prior sensors.
- Machine learning often deploys cascades of circuits, such as a cascade of separate decision trees.
- Each small circuit passes its response to the next circuit, which corrects errors and boosts response quality.
- Learning a sequence of boosted circuits matches the likely way that evolution works, sequentially improving an existing cascade of small modular subsolutions.
- Dimensional reduction provides a potential alternative for anomaly detection. Typical multivariate inputs can be reduced to a lower dimension, similar to principal component analysis. An anomalous input tends to be relatively distant from typical inputs in the reduced space.
- Small encoder circuits can reduce dimensionality, classifying differences in the correlational structure of typical and anomalous inputs. In general, dimensional reduction is likely to be a major feature of biological circuits.
- As in all problems of biological design, evolutionary tuning with respect to tradeoffs inevitably plays a central role in shaping biological circuits.

3. Timescale

3.1. Instantaneous Versus Time-Dependent Inputs

Timescale broadly influences the kinds of circuits that can succeed in anomaly detection. Most anomaly detection methods consider multiple inputs at one point in time. If a single multivariate input is unusual compared with the set of typical multivariate points, then that unusual input is classified as an anomaly.

In some cases, an anomaly must be considered with respect to recent temporal trends [28,29]. For example, reactive oxygen species are often used as weapons in microbial warfare. A rapid increase in concentration of these dangerously reactive molecules may signal an attack.

For multivariate problems that use a single atemporal input, a machine learning method typically classifies by some sort of clustering, partitioning, or dimensional reduction [15–18]. The common inputs fall toward one cluster, or in a particular direction away from a partition, or in a particular location in a reduced space of constructed dimensions. The anomalous inputs are those that are not near the common set.

Temporal problems also require classification [28,29]. However, before classification, one must adjust for the temporal dependence of the input stream. For example, typical inputs may follow a rising trend. An anomaly must be measured against the expected input from the current trend, which requires a circuit to maintain an updated trend estimate.

3.2. Biological Response Times

Atemporal classification of anomalies demands a sufficiently fast circuit. The multivariate perception of input must be accomplished before the environment changes significantly. The calculations to classify must follow with sufficiently short lag to allow an appropriate response.

A neurobiological circuit would likely be quick enough to conduct atemporal classification. For cellular or physiological circuits, response speeds vary widely for different components, from slow biochemical reactions to fast receptors. If the environment changes significantly faster than a circuit's classification inference, then such a circuit may not be able to classify the current environment as if it were an instantaneous isolated event.

Temporal classification over input sequences alters the timescale constraints. The circuit's estimate of trends in inputs may update continuously, although with a time lag. The circuit can work well if its update lag is shorter than the timescale over which environmental trends change.

For temporal classification problems, neurobiological circuits would likely be quick enough for most challenges. Cellular and physiological circuits may sometimes be quick enough if intrinsic temporal smoothing of trend estimation provides sufficient information.

At present, we know little about the cellular and physiological response times of anomaly detection circuits. I limit the discussion to three brief comments.

First, cellular receptors can potentially respond on the timescale of their ligand on–off rates, which are often very fast. So, at the receptor level, sensory information may be able to keep up with environmental change.

Second, some cellular states depend on electric gradients, which change rapidly and can be transmitted at relatively high speed [30]. These fast components of cellular response might provide a sufficient basis for speedy circuits.

Third, slower downstream biochemical reactions might constrain circuit design. Different biochemical processes vary in their response times [31]. Altering the concentrations of reactants often triggers the fastest response. Covalent modifications of enzymes are slower than changes in reactant concentrations. Altering enzyme production or degradation rates is typically the slowest modification of biochemical circuits.

Many other factors could change biochemical response times. However, those factors are likely to be slow relative to the responses of receptors or electric gradients.

4. Simple Mechanisms

4.1. Atemporal Biochemical Mechanism

This subsection briefly illustrates how we may think about mechanistic components in biological circuits. The example describes a simple circuit for atemporal challenges. The following subsection considers temporal challenges.

As a first step, a circuit may determine how each input dimension deviates from its typical value. We begin with a widely observed empirical relation in biochemistry, the Hill function [32–34]. This function describes a common pattern by which an input level is transformed into an output response as

$$h(u|m, k) = \frac{u^k}{m^k + u^k}, \quad (1)$$

in which u is the input, k is a coefficient that determines shape, and m is the input at which the response achieves one-half of its maximum level. The notation $u|m, k$ identifies u as the input variable given the parameters m and k that determine response. In the following, I assume that all functions share the same k value, which is dropped from the notation to simplify the expressions.

We seek a circuit that identifies an anomalous input by its deviation from a standard input level, u^* . Suppose a receptor balances stimulative and repressive forces in relation to input level, given by the difference between two Hill functions

$$\hat{r}(u|m_1, m_2) = h(u|m_1) - a h(u|m_2), \quad (2)$$

in which a is a weighting on the repressive effect to achieve a relative balance between the two forces. We can choose a so that \hat{r} evaluated at u^* is a minimum, creating a circuit that increases in output as the input, u , deviates from the minimum, u^* (see caption for Figure 1).

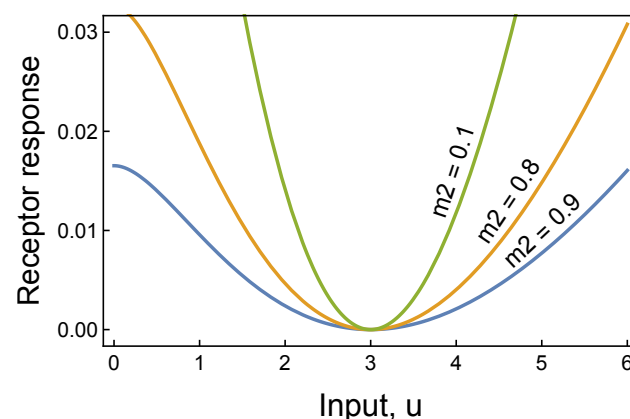


Figure 1. Receptor response for atemporal anomaly detection. When the input is at the typical value of $u^* = 3$, the receptor responds with a minimal value. As the input increasingly deviates from its typical value, the receptor returns an increasing response. The likelihood of an anomalous condition rises with the receptor response value. Thus, this receptor provides a simple atemporal way to classify inputs as normal or anomalous. This figure derives from Equation (3), with $m_1 = 1$, $k = 2$, and a set so that $u^* = 3$ is a minimum. To obtain a minimum at u^* , we search for a such that $d\hat{r}/du = 0$ and $d^2\hat{r}/du^2 > 0$ when evaluated at u^* . For $k = 2$ and $m_1 > m_2$, we obtain $a = (2m_1^2 + u^* + 4)(m_2^2 + u^* + 2)^2 / (2m_2^2 + u^* + 4)(m_1^2 + u^* + 2)^2$. As m_1 and m_2 become large, $a \rightarrow (m_2/m_1)^2$.

For numerical convenience, we can subtract the value of \hat{r} at its minimum to obtain a receptor that returns zero when the input is at its standard level, u^* , and returns increasing values as u increasingly deviates from u^* , as in

$$r(u|m_1, m_2) = \hat{r}(u|m_1, m_2) - \hat{r}(u^*|m_1, m_2). \quad (3)$$

Figure 1 illustrates how this receptor identifies anomalous deviations from typical input values. The cutoff for classifying an anomaly may evolve by natural selection in a biological context or be estimated from data in a machine learning context. Multivariate input requires combining multiple receptor responses to identify anomalies.

4.2. Anomalous Deviation from Temporal Trend

The problem of how cells predict trends has received little attention. This section shows how a very simple biochemical process could in principle be used by cells.

Suppose the value of typical inputs, u^* , changes over time. A circuit must estimate the current typical input. That estimate of u^* and the current input value, u , can be used in Equation (3) to obtain the receptor signal.

If we choose sufficiently large values of the m parameters, then $a \rightarrow (m_2/m_1)^2$ (see caption of Figure 1). With approximately constant a , to calculate the receptor response in Equation (3), we only need to track the dynamics of u^* given the input, u . We can track the dynamics of u^* by including in the circuit

$$\dot{u}^* = \lambda(u - u^*), \quad (4)$$

in which u is a stochastically changing input, and u^* is an exponential moving average of u , with the overdot denoting the derivative with respect to time. The parameter λ controls the speed at which the internal moving average estimate responds to changing inputs. This process simply describes the production or degradation rate of a molecule in response to the level of a stimulating input.

Using the moving average estimate for typical inputs, u^* , in Equation (3) allows the receptor to adjust to changing environmental conditions. Figure 2 shows how this adaptable receptor detects anomalous deviations in the input signal.

In the only similar model that I found, a circuit estimates the ratio of a current input relative to the recent value of typical inputs [35]. The goal was ratio estimation, which the authors called *fold change*. Significant deviations of the ratio from one could also be used for anomaly detection.

To obtain a fold-change circuit, instead of using the receptor in Equation (3), we combine the moving average estimate from Equation (4) with

$$\dot{y} = v\left(\frac{u}{u^*} - y\right), \quad (5)$$

which approximately matches the circuit given published fold-change circuit [35]. If the dynamics respond sufficiently quickly to changes, with large enough λ and v , then y measures the ratio of the current input u to the recent typical input, u^* .

In this ratio-estimating biochemical circuit, the response time may differ from the response time of a receptor-based circuit approximated by Equation (3). Typically, receptors respond more quickly than biochemical reactions. However, it is not clear if natural processes more easily build and tune circuits based on receptors or biochemical reactions in solution.

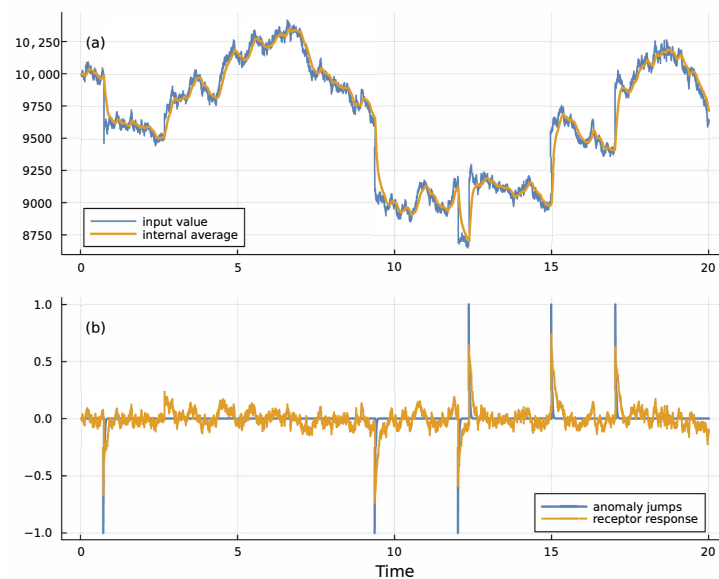


Figure 2. Receptor response for temporal anomaly detection. (a) The blue input signal, u , was generated by a stochastic process $du = 0.0002(10,000 - u) dt + 0.02u dW + zu dN$, in which dW is a Wiener process that generates continuous Gaussian noise with a mean of 0 and a standard deviation of 1, and dN is a Poisson jump process that generates random discrete jumps at rate 0.2. Each jump multiplies the current input, u , by z , which for each jump takes on a value 0.95 or 1.05 with equal probability. The gold moving average, u^* , is given by Equation (4) with $\lambda = 10$. (b) The blue spikes show the timing and direction of the random anomalous jumps for this sample run. The levels of ± 1 for the spikes are arbitrary values. The gold curve shows the receptor output from Equation (3) multiplied by 25, with $k = 2$, $m_1 = 10,000$, $m_2 = 1000$, and a given by the solution in the caption for Figure 1 with $u^* = m_1$. The gold receptor spikes match the blue anomalous input jumps, signaling anomalies. The freely available Julia computer code provides full details about assumptions and methods for all figures in this article [36].

5. Multivariate Signals

The prior subsections analyzed deviations in a single dimension. Detecting anomalies often requires combining information from multiple dimensions. For example, identifying attacks on a computer network depends on the number of data bytes sent to the target computer that may be under attack, the number of data bytes returned to the potential attacker by the target computer, and the type of connection, such as email or web page.

Two widely used test datasets for computer network attack include those network measures along with several other dimensions of data [37,38]. The challenge is to classify whether a network connection to a target computer is a normal use or an attack. Is the connection pattern described by the multivariate measures of the connection typical or anomalous?

Many different machine learning methods have been applied to these benchmark datasets [39–41]. The next subsection begins by evaluating each data dimension independently to infer anomalies and then combining the information in the independent dimensions.

The following subsections analyze the information in the correlation between dimensions by combining the dimensions into a decision tree or by using hierarchical dimension reduction by encoders, two widely used machine learning methods that may map relatively easily to biological circuits.

I start with artificial data to illustrate the methods. I then turn to real data that contrasts typical computer network connections with anomalous connections from attack.

I use the computer data because we do not have large datasets with multivariate measurements of typical and anomalous biological inputs. The goal here is to illustrate the key principles of circuit design that may be important for understanding how natural processes shape biological responses. Anomaly detection has hardly been studied in cellular biology but seems likely to be important in some circumstances.

5.1. Independent Data Dimensions and Ensembles

Suppose an input generates n independent data dimensions. For a typical input, the value in each dimension is a random sample from a normal distribution with mean m_t and standard deviation σ . Similarly, an anomalous input generates n independent values, each sampled from a normal distribution with mean m_a and standard deviation σ . Assume typical inputs are usually smaller than anomalous inputs, $m_t < m_a$.

Suppose a biological circuit can average the n independent values associated with each input. Then, the standard deviation of the average value is σ/\sqrt{n} . The circuit classifies the average value as typical if it is less than a threshold value, τ , and anomalous if greater than the threshold.

Figure 3a illustrates how a change in threshold value alters the circuit's success at classifying inputs. A smaller threshold causes a higher rate of classification as anomalous, which increases both the true rate of predicting anomalies and the false rate of predicting anomalies. As the threshold changes, the curve traces the tradeoffs between those different aspects of successful classification. The area under the curve (AUC) provides one way to measure the overall quality of the circuit's ability to classify inputs.

Figure 3b shows how the circuit's response characteristics improve for increasing levels of n , the number of data dimensions sampled by the circuit. More data dimensions provide more precise information about whether the input is typical or anomalous.

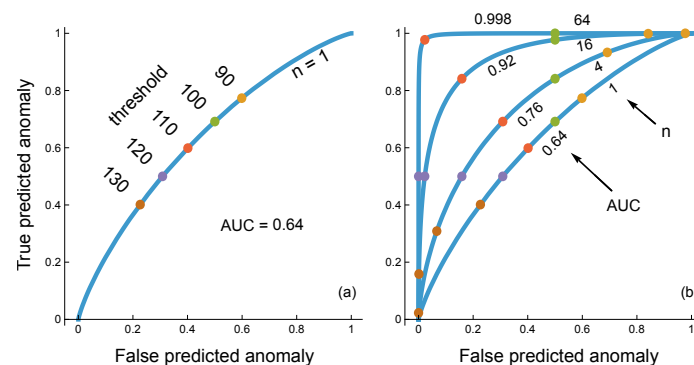


Figure 3. Classification of input as typical or anomalous by a circuit that averages n independent input values and makes a decision based on the average value. Inputs are continuous numerical values. In this example, I generated inputs by randomly sampling a normal distribution with a standard deviation of $\sigma = 40$. For typical and anomalous inputs, the distribution means are 100 and 120, respectively. (a) The circuit takes $n = 1$ dimensions of input. The circuit uses a threshold, τ , such that the circuit classifies inputs below the threshold as typical and above the threshold as anomalous. The curve plots the frequency of truly predicted anomalies as a function of τ versus the frequency of falsely predicted anomalies as a function of τ , generating a receiver operating characteristic (ROC) curve. The area under the curve (AUC) measures the resolving power of the circuit that describes the tradeoff between true positive and false positive classifications over all of the thresholds. (b) For each case in which the true generating process is either typical or anomalous, I generated n independent samples for the associated probability distribution. The circuit measures the average of the inputs, which, when compared to the $n = 1$ case in the left panel, has the same mean and a reduced standard deviation, $\sigma/\sqrt{n} = 40/\sqrt{n}$. The reduced variation provides the circuit with greater resolving power, described by the increasing AUC with increasing n .

5.2. Digital Circuits

Precise estimates for each of the n data values may be difficult for biological sensors, making the circuit sensitive to perturbations in measurement. Suppose instead that each sensor encoded its response in a binary way, which we can label as 0 or 1. In other words, each sensor converts its analog input to a digital output: a 0 response when the value is below some threshold and a 1 response above the threshold. Such analog to digital conversion can be approximated by the Hill function response described by Equation (1), which is widely observed in biology [32–34].

With digital sensors, a circuit only has to combine the information into an overall frequency of 1 values, which are the anomaly signals. For example, if each sensor can trigger the activation of a transcription factor, then those transcription factors can bind to a gene promoter. By this process, the promoter can produce a response that grades with the overall frequency of anomaly signals from the sensors.

This digital circuit requires two threshold values. First, τ sets the point below which an individual sensor returns 0 for a typical input and above which the sensor returns 1 for an anomalous input. Second, a threshold ϕ sets the frequency of 1 responses among the individual sensors required for the circuit to return an overall classification of anomalous for a multivariate input.

Figure 4 shows how the two thresholds interact. Higher curves correspond to increasing numbers of sensors, n . In (a), with $\phi = 1/3$, low thresholds for the individual sensors, τ , cause increasing n to provide relatively high false predicted anomalies (false positives). This pattern can be seen by starting with the lower curve for $n = 1$ and the smallest labeled threshold of 90 marked by the gold circle.

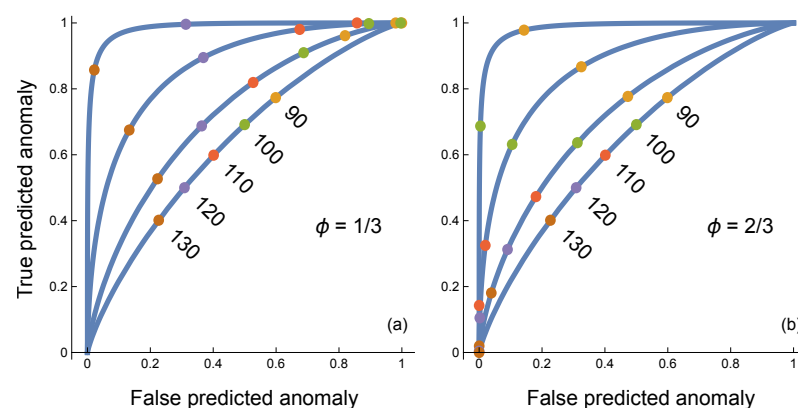


Figure 4. Anomaly classifier in which each sensor conducts an analog to digital conversion, with $n = 1, 4, 16, 64$ sensors for curves from bottom to top. The generation of continuous input into each sensor is described in the caption of Figure 3. In this case, each sensor receives an independent input value and independently scores its input as 0 for typical or 1 for anomalous based on the threshold shared by all sensors. Colored circles on each curve denote particular threshold values for the individual sensors. The overall classification by the circuit depends on the frequency of 1 values returned by the individual sensors. The circuit returns an anomaly if the frequency of 1 values by individual sensors is greater than $\text{ceiling}(\phi n)/n$, in which the ceiling function returns the smallest integer greater than or equal to its argument. (a) Curves for $\phi = 1/3$. (b) Curves for $\phi = 2/3$. Increasing the frequency threshold, ϕ , lowers both the true and false positive classification rates, which can be seen by comparing the same sensor threshold values between the two panels. When $\phi = 1/2$, the threshold locations, τ , are intermediate between the two panels. The AUCs are 0.64, 0.72, 0.87, 0.99 for curves from bottom to top in both panels. The AUC values for $\phi = 1/2$ are slightly higher in the third significant digit for larger n . Overall, the AUC circuit performance is very flat as a function of varying frequency cutoff, ϕ , over $(1/3, 2/3)$, suggesting that ϕ may be a nearly neutral trait over a wide range in the AUC sense of measuring performance over a range of individual sensor thresholds, τ .

As n increases and the curves rise, the gold circle for 90 moves to the right because the rate of false predicted anomalies along the x -axis increases. The reason is that, with both a low individual sensor threshold and a low overall threshold, the expected outcome for a typical input is a false positive prediction of anomaly. As n increases, the variance declines and the expected outcome increasingly dominates.

In Figure 4b, with $\phi = 2/3$, high thresholds for the individual sensors tend to cause an increase in the inputs being predicted as typical. This increase raises the false positive rate of typical predictions, which corresponds to a reduced level along the y -axis for true predicted anomalies. Once again, as n increases, the variance declines and the expected outcome increasingly dominates, causing a drop in true predicted anomalies.

Figure 5 shows that analog to digital conversion by sensors decreases the maximum available information. The lower blue curve traces the smaller error rate for a fully analog circuit that averages the actual values coming into the sensors, as in Figure 3. The upper gold curve shows the rise in the error rate caused by the information lost to digital conversion, as in Figure 4.

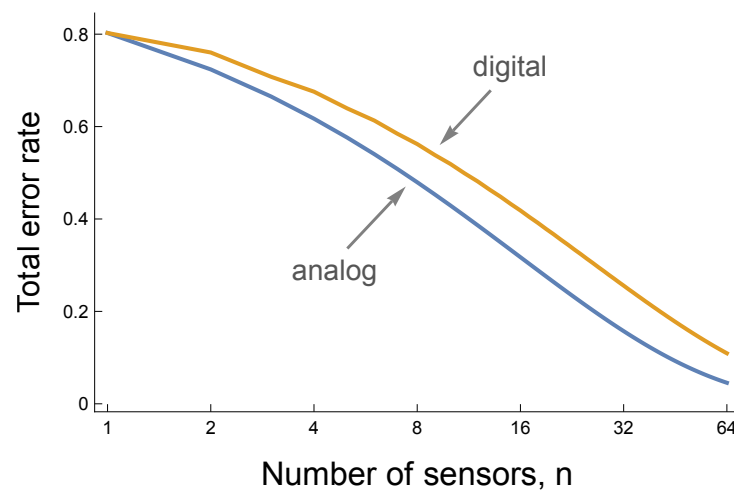


Figure 5. The cost of digitizing the response of individual sensors. The curves show that an increase in the number of sensors, n , reduces the total error rate as the sum of the false negative and false positive rates. In prior figures, the false negative rate is the false predicted anomaly rate, and the false positive rate is one minus the true predicted anomaly rate. The lower blue analog curve corresponds to a circuit that averages the values perceived by the n individual sensors. The upper gold digital curve corresponds to a circuit in which each sensor transforms its input into a 0 response when the input value is below a sensor-specific threshold and a 1 response otherwise. For a given number of sensors, n , the digital circuit produces more errors because digitization at the individual sensor level loses information.

Digital circuits reduce information but are simpler to construct and often are more robust. Small perturbation will usually not alter the 0/1 classification by a sensor. By contrast, many sources of noise will cause variability in a measured analog value.

5.3. Computer Network Anomaly Detection

In the NSL-KDD dataset of attacks on a central computer, a digital ensemble of sensors performs very well at detecting anomalous computer network characteristics associated with attacks. This dataset is widely used as a benchmark for machine learning studies of anomaly detection. The dataset contains measurements for many features of the computer network [38].

A freely available Python (NSL-KDD-01-EDA-OneR: 0.929 ROC-AUC, version 3, 2022) notebook calculated how well each of 36 features could independently classify an input as

a typical network pattern or an anomalous attack [42]. The analysis used the area under the curve (AUC) to measure the resolving power of a feature, as in Figure 3. Features with high resolving power included the amount of data sent by the remote computer to the target computer, the amount of data returned to the remote computer, the kind of service request to the target, such as email or web page, and the number of recent connections by the same remote computer.

The AUC values for 22 of 36 features were greater than 0.5, which means those features had some resolving power. The top 15 AUC values ranged from 0.82 to 0.66. If each sensor's response is encoded as 0 for typical and 1 for anomalous, then an ensemble digital analysis can be created by summing the values for the 22 resolving features. The ensemble circuit's AUC score is 0.93, which is good.

F1 provides another measure of classification success, combining how often a positive prediction is correct and how often a positive input is correctly predicted [43]. The ensemble circuit's F1 score is 0.9, which is also good.

Reducing the number of sensors to the top 4 with individual AUC values above 0.75, the ensemble AUC score is 0.92, and the F1 score is 0.89. Thus, a small and simple ensemble of digital sensors performs well for this classic benchmark dataset.

5.4. Extra Information in Multivariate Pattern

In the ensemble digital model, each sensor passes a digital response. That response can easily be combined with the outputs of other digital sensors to create an overall circuit response. Simple biological circuits may often be built in this way.

The digital ensemble uses each dimension of the input independently. Each digital sensor takes one input value and responds as a one-step decision tree (Figure 6a). If the input is greater than some threshold, the decision tree responds one way. Otherwise, it responds the other way.

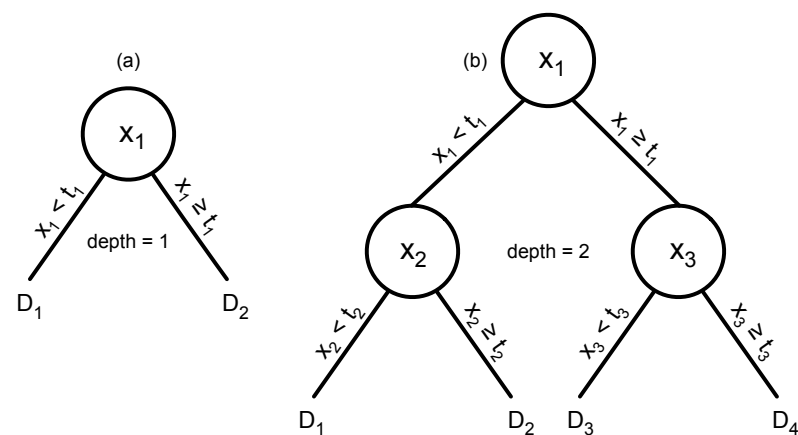


Figure 6. Decision trees for an anomaly detection classification problem. The challenge is to classify a multivariate data input with values x_i for the $i = 1, \dots, N$ data dimensions. (a) A tree of depth 1 that predicts classification based on one feature of a multivariate observation, x_1 . If x_1 is greater than or equal to a threshold, t_1 , then the tree returns a decision value, D_2 . Otherwise, it returns D_1 . If there is only a single tree, then the decision value determines the classification. Alternatively, there may be an ensemble of trees, each tree analyzing a different data dimension. In an ensemble, each tree contributes a separate decision value that can be combined with the values from other trees to make an overall classification decision. (b) A tree of depth 2 that uses three different data features.

However, a multivariate pattern rarely occurs as a collection of independent dimensions. Most machine learning methods extract some of the extra multivariate information. The following sections consider two common machine learning circuits that may apply widely in biology.

6. Boosted Decision Trees

6.1. Deep Trees

A simple extension uses deeper decision trees. In Figure 6b, the input value for the first feature of the multivariate data is split at some threshold value. If the first feature is greater than its threshold, then a second split occurs based on another feature and a different threshold. If the first feature is less than its threshold, then the second split happens based on different criteria.

A deeper tree analyzes multiple inputs, allowing for decisions that include correlations between different feature dimensions of the data. A tree of depth n creates $2^n - 1 \approx 2^n$ splits in the data. For example, if a system has the capacity to create $2^6 = 64$ splits, then it can create $2^0 = 1$ trees of depth 6, or $2^1 = 2$ trees of depth 5, or $2^2 = 4$ trees of depth 4, and so on.

Approximately, for 2^n splits, the system can create 2^m trees of depth 2^{n-m} . Typically, machine learning applications perform better by using many trees of shallower depth rather than a small number of deep trees. Various methods exist for creating multiple trees and combining them into a single decision ensemble [44,45].

6.2. Boosting and Biological Design

The most widely successful method creates trees by a boosting process [46]. Boosting creates trees sequentially, starting with a single relatively small tree. Then, with an optimized first tree, the algorithm adds a second tree that corrects errors made by the first tree. The process continues adding trees in this way, each tree boosting the success achieved by the prior ensemble.

Boosting seems like a good description of how biological circuits may be designed by natural selection. Initially, a small circuit may provide some information. A second circuit may boost performance, followed by a third, and so on. Sequentially boosted improvement may be the essence of biological design.

6.3. Typical vs. Anomalous Data as Self vs. Nonself

Figure 7 illustrates some of the tradeoffs in building an ensemble of boosted trees. In this case, I generated an artificial set of data with both typical and anomalous inputs by sampling from multivariate normal distributions. Each input has f feature dimensions.

For the typical data, each feature dimension has a mean value drawn randomly from a normal distribution with mean zero and standard deviation σ . I call that standard deviation the mean scale because σ determines the scale of the fluctuations among the means of the different dimensions.

The variance in each dimension is one, so the f -dimensional correlation matrix is also the covariance matrix. I generated that matrix by a random draw from a uniform distribution over all possible correlation matrices [47]. Once this distribution is set for typical data, all typical observations come from this single distribution.

For the anomalous data, I used the same process to create a new multivariate normal distribution for each observation. Each anomalous observation is a single random draw from a unique distribution. Thus, classification requires recognizing what a typical observation looks like when compared with a wide variety of anomalous data patterns rather than recognizing specific signatures of danger. This structure captures the essence of self versus nonself discrimination. Here, the typical pattern defines self, and the anomalous observations define nonself, the variety of patterns distinct from self [48–53].

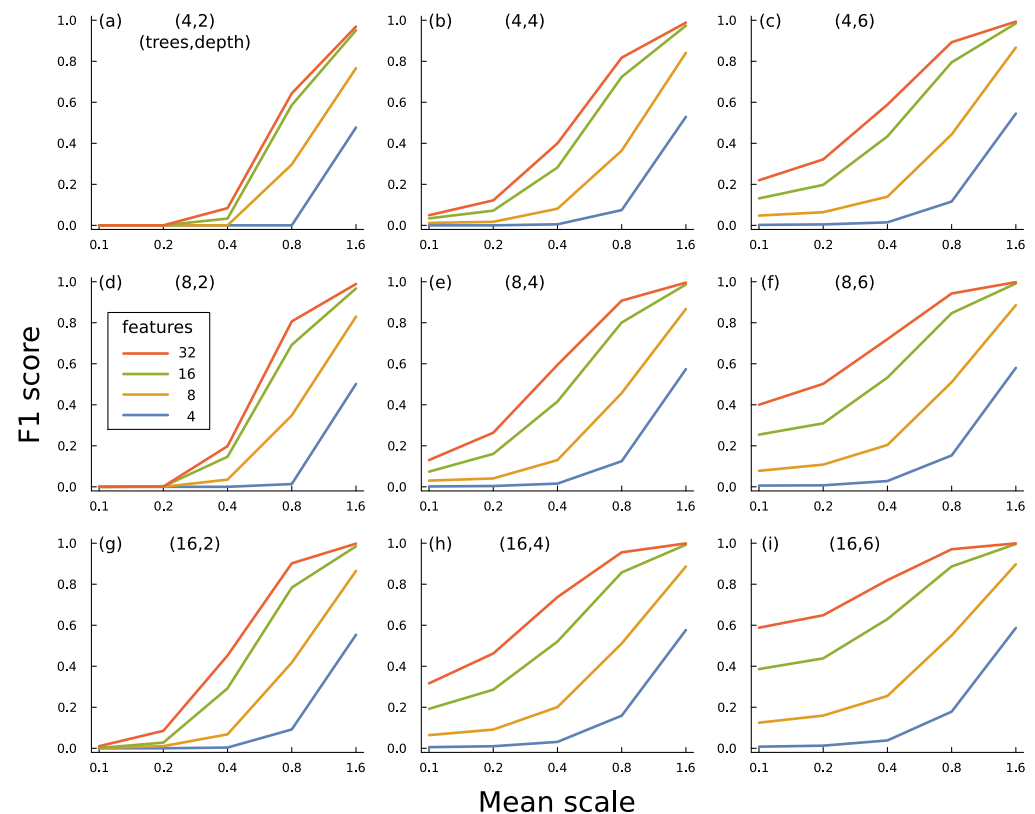


Figure 7. Performance of boosted decision tree ensembles for classifying typical versus anomalous inputs. (a–i) Mean scale influences the amount of deviation in mean values between typical and anomalous inputs. F1 score measures the success of a circuit in classifying typical and anomalous data [43]. That score combines how often a prediction of anomaly is correct with how often an anomalous input is correctly identified. Features is the number of dimensions in the data. Trees is the number of trees in an ensemble circuit. Depth is the depth of each tree in a circuit. The text describes the methods and main conclusions for this figure. I generated one dataset with 32 features and used subsets of the feature data for the various plots so that the correlation structure of the data was consistent between the various comparisons. The boosted tree ensemble was calculated by the widely used xgboost algorithm [54]. For T trees each of depth n , the total number of splits is $(2^n - 1)T$.

6.4. Performance

Figure 7a shows the success of a boosted decision tree circuit. For that panel, the circuit has 4 trees, each of depth 2. As the mean scale increases along the x -axis, the circuits improve at detecting anomalies. A greater mean scale implies that, for each feature, the average deviation between the mean values of the typical and anomalous observations rises. Decision trees can easily detect differences in mean values for a feature by splitting at a threshold that likely separates typical and anomalous inputs.

The different curves show the varying numbers of features available in the data. More features tend to increase the largest deviations in mean values between typical and anomalous observations. More features also tend to increase the difference in multivariate correlation structure between typical and anomalous observations because greater dimensionality increases the space of possible correlation patterns.

The other panels show the increase in classification success as the number of trees or the depth of trees increases. Deeper trees are particularly good at identifying differences in multivariate patterns caused by correlations between features. That benefit can be seen by comparing the success of the deeper trees at low values of mean scale, for which there is little information available from differences in mean values between typical and anomalous observations.

The structure of this particular problem provides a strong challenge for anomaly detection because no common pattern exists among the anomalous inputs. Additionally, the generating process for the observations creates wide scatter among both typical and anomalous inputs. Nonetheless, the boosted tree ensembles significantly discriminate between typical and anomalous inputs.

6.5. Boosted Trees and Biological Circuit Evolution

Each node of a tree is simply a binary split based on input. Thus, any biological circuit that expresses the commonly observed Hill response could implement a node of a decision tree [32–34]. Combining information from multiple trees is also likely to be something that simple biological systems could achieve.

As I mentioned earlier, the sequential process of building boosted trees likely matches the natural tendency for evolutionary processes to create solutions by adding improvements to an initial design. Thus, the simple way in which tree-like decision nodes can be implemented biologically and the sequential process of boosting make boosted trees an excellent model for cellular and neural circuits that solve challenges of classification and decision.

7. Encoders and Internal Models

7.1. Dimensional Reduction

Encoders reduce dimension by compressing inputs into informative components (for background, see Box 2 of Frank [19]). Dimensional reduction by encoding can be an effective way to identify anomalous environmental conditions. A common autoencoder method first compresses the f features of an input to a representation in a lower f' -dimensional space. It then expands that representation back to the original f dimensions, attempting to match the original input closely.

An autoencoder uses patterns in the data [41]. For example, suppose the second feature tends to follow a particular function of the third and fourth features. In that case, the compression method can discard the second feature and recreate that feature during decompression. When a good autoencoder compresses and then decompresses an input, the final decompressed value tends to be close to the original input.

If anomalous inputs lack some of the patterns in typical inputs, an autoencoder built for typical inputs will often distort an anomalous input during the encoding–decoding sequence. The output for an anomalous input will often be farther from the original input than usually happens for typical inputs. Thus, the distance between the input and the output of an autoencoder can be used to classify inputs as typical or anomalous.

Using a sequence of compression steps often creates a more effective encoding. If, for example, the initial data have 2^n features, a first compression stage may reduce to 2^{n-1} dimensions, followed by compression to 2^{n-2} dimensions, and so on. Sequential compression helps to create an internal model of the data [18,55–57]. When sequentially compressing images of faces, early steps may focus on facets such as eyes, ears, nose, and mouth. Later steps may consider relations between those parts, providing an internal model of how a typical face tends to look [58–60].

7.2. Anomaly Detection

For this article, we are particularly interested in the simplest effective circuits. A full autoencoder requires both encoding compression and decoding decompression. A simpler approach uses only the encoding step. We convert the f features in the input to f' compressed dimensions. If we design an encoder that tends to create a large distance between typical and anomalous inputs in the f' -dimensional representation, then we can use that distance to detect anomalies.

Figure 8 illustrates how an encoder separates typical and anomalous inputs. In this example, the four input dimensions were reduced to two output dimensions using a single-layer neural network. That small network separated typical and anomalous observations with high accuracy.

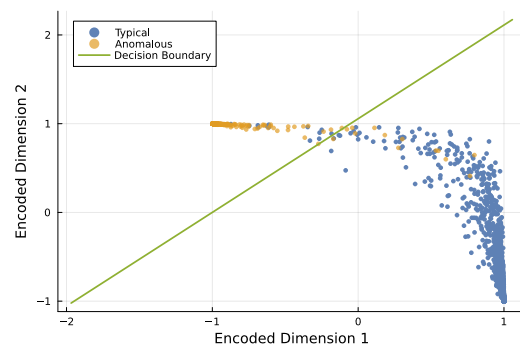


Figure 8. Encoder model that reduces 4-dimensional inputs to 2 dimensions, separating typical and anomalous observations. I used the same methods to generate the data as for boosted trees, described previously. Of the initial 100,000 observations, 10% are anomalies, and the rest are typical. I randomly split the data into a training set composed of 70% of the observations and the remainder in the test set to evaluate the fitted model. This plot shows a random subset of the test data with approximately 2700 typical observations and 300 anomalous observations. Compared with Figure 9, the mean scale value here is 1.6, and the number of features is 4. I used a distinct dataset for this figure to provide a visualization that shows the separation between typical and anomalous points more clearly. In this case, the F1 score is 0.96, corresponding to relatively few misclassified points. The model encoded the 4 input dimensions to 2 output dimensions with a single layer of a neural network using 10 parameters.

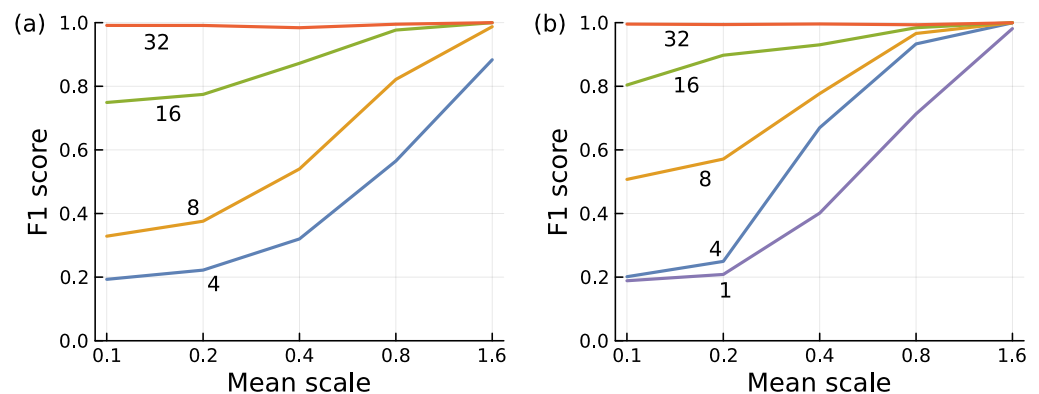


Figure 9. Encoder model to separate typical from anomalous inputs. The labels on each curve describe the number of features, f , in the data. I used the same methods to generate the data as for boosted trees, described previously. (a) I generated three separate input datasets and calculated F1 scores for each to compensate for peculiarities of any particular dataset. I then averaged the three values for each mean scale by feature combination. The overall pattern and magnitudes for each separate dataset were similar. For each dataset, I generated 32 features. I then used the first f features in the set for each curve. If we compress from inputs with $f = 2^n$ feature dimensions to a single output dimension, then a full encoder model has $(2f + 5)(f - 1)/3$ parameters when n is an integer. (b) I began calculation for each point using all 32 features. I then iteratively deleted the single feature that provided the least information, measured by the smallest reduction in F1 when deleting that feature. I continued until the specified number of features for a particular curve remained. This iterative deletion method provided a better set of features than simply picking the first f features as in (a).

7.3. Factors Influencing Circuit Accuracy

Figure 9a compares an encoder's classification efficacy under different conditions. The F1 score measures classification efficacy, combining how often a positive prediction is correct and how often a positive input is correctly predicted [43]. The mean scale influences the amount of variation between typical and anomalous mean values in each dimension of the data.

The full data initially contained $f = 32$ features. I then calculated F1 scores by using only the first $f = 4, 8, 16$ feature dimensions. Each line in the figure is labeled with the number of features used, $f = 2^n$. For this figure, an encoder reduces dimensionality from the $f = 2^n$ inputs to $2^0 = 1$ output, using n layers in the neural network encoder.

Three conclusions follow from this figure. First, between typical and anomalous inputs, larger differences in mean values for each dimension make it easier to detect anomalies, shown in the figure as the mean scale increases along the x -axis.

Second, sampling more features of the data improves classification. The improvement occurs primarily for small values of the mean scale, in which mean differences provide limited information. In those situations, a classifier can succeed when it is able to infer distinctions between typical and anomalous inputs in the correlation pattern among the dimensions. In this example, raising the number of features enhances the information about correlational pattern, providing increasingly accurate classification.

The third conclusion is that, given a sufficient number of features, an encoder can achieve nearly perfect classification for these input data. In this case, an encoder using all 32 features of the data made very few classification errors.

The encoder for $f = 32$ features achieved high success by optimizing the 713 parameters in its encoding network. For $f = 4, 8, 16$, the circuits required 13, 49, and 185 parameters.

7.4. Simplifying Circuits

Figure 9b shows that, for a given performance level measured by F1, we can find simpler circuits with the same performance. In that plot, the calculation of each point began with all 32 features. I then iteratively removed one feature at a time, dropping the feature that provided the least amount of information, measured by the smallest decline in F1. I continued dropping features in this way, providing an F1 measure for $f = 1, 2, \dots, 32$ for each mean scale level. The plot shows curves for particular f values.

Choosing the best f features of the full 32 in the data provides a better F1 score than taking the first f features in the data, as expected. Put another way, for a given F1 level of performance, we can use a smaller circuit if we select the best features rather than using a fixed feature set. The amount by which a circuit can be reduced for a given F1 score depends on the particular data structure, as shown by the plots.

We could further reduce the number of parameters in a circuit by imposing a cost on each parameter. A cost favors a parameter to decline close to zero when it adds relatively little improvement in performance. We then obtain a simplified circuit by pruning all parameters near zero.

Overall, relatively small encoder circuits can achieve good classification for some types of data.

8. Discussion

I have focused on anomalies as unusual observations, anything that differs from what is typical. Detection does not depend on specific anomalous patterns or danger signals. Instead, a system creates a model of a typical input and infers when an input differs

from that internal model, something like “That’s an unusual smell” or “I’ve never seen that before”.

Sensory or neural adaptation provides a simple example. Many biological circuits adjust their baseline by averaging over recent inputs. That baseline allows the circuit to measure deviations from what has recently been typical [61–65]. I presented simple circuits in Equations (4) and (5) that adapt to recent trends. An ensemble of such circuits could classify multivariate inputs.

Self versus nonself recognition occurs widely throughout biology [66–75]. In some cases, nonself is recognized by direct pattern recognition, which does not require the more challenging kinds of circuits discussed in this article. In other cases, self recognition is more complex and not fully understood [12]. It seems that systems sometimes recognize what is self and classify as anomalous those observations that do not fit the self pattern, potentially sharing properties with the machine-learning circuits discussed in this article.

The human hippocampus appears to recognize novelty in certain contexts [7–9]. Further studies suggest that memory creates a model of what is common. The system classifies inputs as novel or unusual when they deviate significantly from expectations [76,77]. With regard to the analyses in this article, some sort of dimensional reduction likely encodes the internal model.

Cellular and physiological systems would likely gain from anomaly detection. The models in this article suggest the kinds of small circuits that could work within these constrained biological systems.

Funding: The Donald Bren Foundation, US Department of Defense grant W911NF2010227, and US National Science Foundation grant DEB-2325755 support my research. The APC was funded by the Donald Bren Foundation.

Data Availability Statement: Software to produce the figures is available on GitHub [36].

Acknowledgments: Declaration of generative AI and AI-assisted technologies in the writing process. During the preparation of this work, the author used several OpenAI models in order to research background information on various topics and search for references in the literature. After using this tool/service, the author reviewed and edited the content as needed and takes full responsibility for the content of the publication.

Conflicts of Interest: The author declares no conflicts of interest.

References

1. Heil, M.; Land, W.G. Danger signals—Damaged-self recognition across the tree of life. *Front. Plant Sci.* **2014**, *5*, 107937. [[CrossRef](#)] [[PubMed](#)]
2. LeRoux, M.; Peterson, S.B.; Mougous, J.D. Bacterial danger sensing. *J. Mol. Biol.* **2015**, *427*, 3744–3753. [[CrossRef](#)] [[PubMed](#)]
3. Nesse, R.M. The smoke detector principle: Signal detection and optimal defense regulation. *Evol. Med. Public Health* **2019**, *2019*, 1. [[CrossRef](#)] [[PubMed](#)]
4. Klein, A.S.; Dolensek, N.; Weiand, C.; Gogolla, N. Fear balance is maintained by bodily feedback to the insular cortex in mice. *Science* **2021**, *374*, 1010–1015. [[CrossRef](#)]
5. Moscarello, J.M.; Penzo, M.A. The central nucleus of the amygdala and the construction of defensive modes across the threat-imminence continuum. *Nat. Neurosci.* **2022**, *25*, 999–1008. [[CrossRef](#)]
6. Doyle, A.C. The adventure of Silver Blaze. In *The Memoirs of Sherlock Holmes*; George Newnes Ltd.: London, UK, 1893.
7. Kumaran, D.; Maguire, E.A. An unexpected sequence of events: Mismatch detection in the human hippocampus. *PLoS Biol.* **2006**, *4*, e424. [[CrossRef](#)]
8. Kumaran, D.; Maguire, E.A. Match–mismatch processes underlie human hippocampal responses to associative novelty. *J. Neurosci.* **2007**, *27*, 8517–8524. [[CrossRef](#)]
9. Bhasin, G.; Nair, I.R. Dynamic hippocampal CA2 responses to contextual spatial novelty. *Front. Syst. Neurosci.* **2022**, *16*, 923911. [[CrossRef](#)]

10. Dasgupta, D.; Forrest, S. An anomaly detection algorithm inspired by the immune system. In *Artificial Immune Systems and Their Applications*; Dasgupta, D., Ed.; Springer: New York, NY, USA, 1999; pp. 262–277.
11. Ramadan, A.; Land, W.G.; Paczesny, S. Editorial: Danger signals triggering immune response and inflammation. *Front. Immunol.* **2017**, *8*, 979. [\[CrossRef\]](#)
12. Koncz, B.; Balogh, G.M.; Manczinger, M. A journey to your self: The vague definition of immune self and its practical implications. *Proc. Natl. Acad. Sci. USA* **2024**, *121*, e2309674121. [\[CrossRef\]](#)
13. Meents, A.K.; Mithöfer, A. Plant–plant communication: Is there a role for volatile damage-associated molecular patterns? *Front. Plant Sci.* **2020**, *11*, 583275. [\[CrossRef\]](#)
14. Duc, N.H.; Vo, H.T.; van Doan, C.; Hamow, K.Á.; Le, K.H.; Posta, K. Volatile organic compounds shape belowground plant–fungi interactions. *Front. Plant Sci.* **2022**, *13*, 1046685. [\[CrossRef\]](#) [\[PubMed\]](#)
15. Omar, S.; Ngadi, A.; Jebur, H.H. Machine learning techniques for anomaly detection: An overview. *Int. J. Comput. Appl.* **2013**, *79*, 33–41. [\[CrossRef\]](#)
16. Nassif, A.B.; Talib, M.A.; Nasir, Q.; Dakalbab, F.M. Machine learning for anomaly detection: A systematic review. *IEEE Access* **2021**, *9*, 78658–78700. [\[CrossRef\]](#)
17. Pang, G.; Shen, C.; Cao, L.; Hengel, A.V.D. Deep learning for anomaly detection: A review. *ACM Comput. Surv. (CSUR)* **2021**, *54*, 1–38. [\[CrossRef\]](#)
18. Ruff, L.; Kauffmann, J.R.; Vandermeulen, R.A.; Montavon, G.; Samek, W.; Kloft, M.; Dietterich, T.G.; Müller, K.R. A unifying review of deep and shallow anomaly detection. *Proc. IEEE* **2021**, *109*, 756–795. [\[CrossRef\]](#)
19. Frank, S.A. Circuit design in biology and machine learning. I. Random networks and dimensional reduction. *arXiv* **2025**, arXiv:2408.09604. [\[CrossRef\]](#)
20. De Jong, K.A. *Evolutionary Computation: A Unified Approach*, 2nd ed.; MIT Press: Cambridge, MA, USA, 2006.
21. Holland, J.H. *Adaptation in Natural and Artificial Systems*; University of Michigan Press: Ann Arbor, MI, USA, 1975.
22. Holland, J.H. *Hidden Order: How Adaptation Builds Complexity*; Basic Books: New York, NY, USA, 1995.
23. McCulloch, W.S.; Pitts, W. A logical calculus of ideas immanent in nervous activity. *Bull. Math. Biophys.* **1943**, *5*, 115–133. [\[CrossRef\]](#)
24. Hebb, D.O. *The Organization of Behavior*; Wiley: New York, NY, USA, 1949.
25. Rumelhart, D.E.; McClelland, J.L. *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Volume 1: Foundations*; MIT Press: Cambridge, MA, USA, 1986.
26. Jiahui, G.; Feilong, M.; Visconti di Oleggio Castello, M.; Nastase, S.A.; Haxby, J.V.; Gobbini, M.I. Modeling naturalistic face processing in humans with deep convolutional neural networks. *Proc. Natl. Acad. Sci. USA* **2023**, *120*, e2304085120. [\[CrossRef\]](#)
27. Kanwisher, N.; Khosla, M.; Dobs, K. Using artificial neural networks to ask ‘why’ questions of minds and brains. *Trends Neurosci.* **2023**, *46*, 240–254. [\[CrossRef\]](#)
28. Blázquez-García, A.; Conde, A.; Mori, U.; Lozano, J.A. A review on outlier/anomaly detection in time series data. *ACM Comput. Surv. (CSUR)* **2021**, *54*, 1–33. [\[CrossRef\]](#)
29. Choi, K.; Yi, J.; Park, C.; Yoon, S. Deep learning for anomaly detection in time-series data: Review, analysis, and guidelines. *IEEE Access* **2021**, *9*, 120043–120065. [\[CrossRef\]](#)
30. Levin, M. Molecular bioelectricity: How endogenous voltage potentials control cell behavior and instruct pattern regulation in vivo. *Mol. Biol. Cell* **2014**, *25*, 3835–3850. [\[CrossRef\]](#) [\[PubMed\]](#)
31. Fell, D. *Understanding the Control of Metabolism*; Portland Press: London, UK, 1997.
32. Frank, S.A. Input-output relations in biological systems: Measurement, information and the Hill equation. *Biol. Direct* **2013**, *8*, 31. [\[CrossRef\]](#)
33. Zhang, Q.; Bhattacharya, S.; Andersen, M.E. Ultrasensitive response motifs: Basic amplifiers in molecular signalling networks. *Open Biol.* **2013**, *3*, 130031. [\[CrossRef\]](#)
34. Martinez-Corral, R.; Nam, K.M.; DePace, A.H.; Gunawardena, J. The Hill function is the universal Hopfield barrier for sharpness of input–output responses. *Proc. Natl. Acad. Sci. USA* **2024**, *121*, e2318329121. [\[CrossRef\]](#)
35. Goentoro, L.; Shoval, O.; Kirschner, M.W.; Alon, U. The incoherent feedforward loop can provide fold-change detection in gene regulation. *Mol. Cell* **2009**, *36*, 894–899. [\[CrossRef\]](#)
36. Frank, S.A. Circuit Design in Biology and Machine Learning. II. Julia Software Code. 2024. Available online: https://github.com/evolbio/Circuits_02/releases/tag/Version_1.0.0 (accessed on 1 July 2025).
37. Ring, M.; Wunderlich, S.; Landes, D.; Hotho, A. CIDDs-001. 2022. Available online: <https://www.kaggle.com/datasets/dhoogla/cidds001> (accessed on 1 July 2025).
38. Tavallaee, M.; Bagheri, E.; Lu, W.; Ghorbani, A. NSL-KDD. 2022. Available online: <https://www.kaggle.com/datasets/dhoogla/nslkdd> (accessed on 1 July 2025).
39. Thapa, N.; Liu, Z.; KC, D.B.; Gokaraju, B.; Roy, K. Comparison of machine learning and deep learning models for network intrusion detection systems. *Future Internet* **2020**, *12*, 167. [\[CrossRef\]](#)

40. Salih, A.A.; Abdulazeez, A.M. Evaluation of classification algorithms for intrusion detection system: A review. *J. Soft Comput. Data Min.* **2021**, *2*, 31–40. [\[CrossRef\]](#)
41. Torabi, H.; Mirtaheeri, S.L.; Greco, S. Practical autoencoder based anomaly detection by using vector reconstruction error. *Cybersecurity* **2023**, *6*, 1. [\[CrossRef\]](#)
42. D’Hooge, L. *NSL-KDD-01-EDA-OneR: 0.929 ROC-AUC*, version 3; The Python Software Foundation: Wilmington, DE, USA, 2022.
43. Powers, D.M.W. Evaluation: From precision, recall and F-measure to ROC, informedness, markedness and correlation. *arXiv* **2020**, arXiv:2010.16061. [\[CrossRef\]](#)
44. Zhou, Z.H. *Ensemble Methods: Foundations and Algorithms*; CRC Press: Boca Raton, FL, USA, 2012.
45. Hastie, T.; Tibshirani, R.; Friedman, J. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed.; Springer: New York, NY, USA, 2017.
46. Schapire, R.E.; Freund, Y. Boosting: Foundations and algorithms. *Kybernetes* **2013**, *42*, 164–166. [\[CrossRef\]](#)
47. Lewandowski, D.; Kurowicka, D.; Joe, H. Generating random correlation matrices based on vines and extended onion method. *J. Multivar. Anal.* **2009**, *100*, 1989–2001. [\[CrossRef\]](#)
48. Aickelin, U.; Dasgupta, D.; Gu, F. Artificial immune systems. In *Search Methodologies*; Burke, E., Kendall, G., Eds.; Springer: Berlin/Heidelberg, Germany, 2014; pp. 187–211.
49. Dasgupta, D.; Forrest, S. Artificial immune systems in industrial applications. In Proceedings of the Second International Conference on Intelligent Processing and Manufacturing of Materials, IPMM’99, Honolulu, HI, USA, 10–15 July 1999; pp. 257–267.
50. Forrest, S.; Perelson, A.S.; Allen, L.; Cherukuri, R. Self-nonsel self discrimination in a computer. In Proceedings of the Proceedings of 1994 IEEE Computer Society Symposium on Research in Security and Privacy, Oakland, CA, USA, 16–18 May 1994; pp. 202–212.
51. Hofmeyr, S.A.; Forrest, S. Architecture for an artificial immune system. *Evol. Comput.* **2000**, *8*, 443–473. [\[CrossRef\]](#)
52. Kim, J.; Bentley, P.J. An evaluation of negative selection in an artificial immune system for network intrusion detection. In Proceedings of the 3rd Annual Conference on Genetic and Evolutionary Computation, San Francisco, CA, USA, 7–11 July 2001; Morgan Kaufmann Publishers Inc.: San Francisco, CA, USA, 2001; pp. 1330–1337.
53. Stibor, T.; Timmis, J.; Eckert, C. A comparative study of real-valued negative selection to statistical anomaly detection techniques. In Proceedings of the International Conference on Artificial Immune Systems, Banff, AB, Canada, 14–17 August 2005; Springer: Berlin/Heidelberg, Germany, 2005; pp. 262–275.
54. Chen, T.; Guestrin, C. Xgboost: A scalable tree boosting system. In Proceedings of the KDD ’16: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 785–794.
55. Bengio, Y. Learning deep architectures for AI. *Found. Trends Mach. Learn.* **2009**, *2*, 1–127. [\[CrossRef\]](#)
56. Bengio, Y.; Courville, A.; Vincent, P. Representation learning: A review and new perspectives. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 1798–1828. [\[CrossRef\]](#) [\[PubMed\]](#)
57. Hinton, G.E.; Salakhutdinov, R.R. Reducing the dimensionality of data with neural networks. *Science* **2006**, *313*, 504–507. [\[CrossRef\]](#) [\[PubMed\]](#)
58. Le, Q.V.; Ranzato, M.; Monga, R.; Devin, M.; Chen, K.; Corrado, G.S.; Dean, J.; Ng, A.Y. Building high-level features using large scale unsupervised learning. In Proceedings of the ICML’12: Proceedings of the 29th International Conference on Machine Learning, Edinburgh, UK, 26 June–1 July 2012; Omnipress: Madison, WI, USA, 2012; pp. 507–514.
59. Lee, H.; Grosse, R.; Ranganath, R.; Ng, A.Y. Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In Proceedings of the ICML ’09: Proceedings of the 26th Annual International Conference on Machine Learning, Montreal, QC, Canada, 14–18 June 2009; pp. 609–616.
60. Masci, J.; Meier, U.; Cireşan, D.; Schmidhuber, J. Stacked convolutional auto-encoders for hierarchical feature extraction. In Proceedings of the Artificial Neural Networks and Machine Learning—ICANN 2011, Espoo, Finland, 14–17 June 2011; Honkela, T., Duch, W., Girolami, M., Kaski, S., Eds.; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2011; Volume 6791, pp. 52–59.
61. Carandini, M.; Heeger, D.J. Normalization as a canonical neural computation. *Nat. Rev. Neurosci.* **2012**, *13*, 51–62. [\[CrossRef\]](#)
62. Fairhall, A.L.; Lewen, G.D.; Bialek, W.; van Steveninck, R.R.d.R. Efficiency and ambiguity in an adaptive neural code. *Nature* **2001**, *412*, 787–792. [\[CrossRef\]](#)
63. Kohn, A. Visual adaptation: Physiology, mechanisms, and functional benefits. *J. Neurophysiol.* **2007**, *97*, 3155–3164. [\[CrossRef\]](#)
64. Wark, B.; Lundstrom, B.N.; Fairhall, A. Sensory adaptation. *Curr. Opin. Neurobiol.* **2007**, *17*, 423–429. [\[CrossRef\]](#)
65. Whitmire, C.J.; Stanley, G.B. Adaptation in human and machine perception. *Nat. Neurosci.* **2016**, *19*, 1557–1564.
66. Bedinger, P.A.; Broz, A.K.; Tovar-Mendez, A.; McClure, B. Recognition and rejection of self in plant reproduction. *Science* **2017**, *356*, eaaf5566.
67. Boehm, T. Quality control in self/nonself discrimination. *Cell* **2006**, *125*, 845–858. [\[CrossRef\]](#) [\[PubMed\]](#)
68. Boehm, T.; Rosenstiel, P. Self/nonself recognition across kingdoms: A review of common and unique mechanisms. *Front. Immunol.* **2021**, *12*, 688095.

69. Cooper, E.L. Self/non-self recognition in invertebrates. *Adv. Exp. Med. Biol.* **2010**, *708*, 130–143.
70. De Tomaso, A.W. The evolution of self/non-self recognition in colonial ascidians. *Curr. Opin. Immunol.* **2018**, *51*, 58–64.
71. Mojica, F.J.; Rodriguez-Valera, F. The discovery of CRISPR in archaea and bacteria. *FEBS J.* **2016**, *283*, 3162–3169. [[CrossRef](#)]
72. Pradeu, T. Immunity and the emergence of individuality. In *From Groups to Individuals: Evolution and Emerging Individuality*; Bouchard, F., Huneman, P., Eds.; MIT Press: Cambridge, MA, USA, 2013; pp. 77–96.
73. Ruhe, Z.C.; Low, D.A.; Hayes, C.S. Bacterial contact-dependent growth inhibition. *Trends Microbiol.* **2013**, *21*, 230–237. [[CrossRef](#)]
74. Tock, M.R.; Dryden, D.T. The biology of restriction and anti-restriction. *Curr. Opin. Microbiol.* **2005**, *8*, 466–472. [[CrossRef](#)]
75. Witzany, G.; Nowacki, M. Self-nonsel self recognition in protists and fungi: A review of cellular and molecular mechanisms. *J. Theor. Biol.* **2016**, *399*, 53–63.
76. Frank, D.; Kafkas, A. Expectation-driven novelty effects in episodic memory. *Neurobiol. Learn. Mem.* **2021**, *183*, 107466. [[CrossRef](#)] [[PubMed](#)]
77. Shing, Y.L.; Brod, G.; Greve, A. Prediction error and memory across the lifespan. *Neurosci. Biobehav. Rev.* **2023**, *155*, 105462. [[CrossRef](#)] [[PubMed](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.