Polytopes as vehicles of informational content in feedforward neural networks

Feraz Azhar*

Unit for History and Philosophy of Science, University of Sydney, NSW 2006, Australia

Localizing content in neural networks provides a bridge to understanding the way in which the brain stores and processes information. In this paper, I propose the existence of polytopes in the state space of the hidden layer of feedforward neural networks as vehicles of content. I analyze these geometrical structures from an information-theoretic point of view, invoking mutual information to help define the content stored within them. I establish how this proposal addresses the problem of misclassification, and provide a novel solution to the disjunction problem, which hinges on the precise nature of the causal-informational framework for content advocated herein.

1. Introduction

In attempting to understand how the brain stores and processes information, a natural assumption is that there exist concrete structures within and referenced by the brain that reflect the tasks the brain is built to address. The implicit premise underlying this assumption is that there exists a naturalistic account of representation that will provide an explanation of how the brain performs its functions (Shea, 2013). Though the goal of understanding mental representation has remained elusive, recent advances in neuroscience provide clues as to how we might proceed. We need not yet be overwhelmed by the complexity exhibited by the $\sim 10^{11}$ neurons in the brain, each with $\sim 10^{4}$ synaptic connections, as bottom-up approaches, which analyze networks from the point of view of single units, reveal the existence of complex and varied computational capacity (Rieke et al., 1997). This provides support for a program that promotes the study of individual units arranged in architecturally realistic networks, with the goal of identifying potential representational structures the brain uses in its functioning, understanding how they come about, and what relationships they enter into with respect to each other and the external world.

The motivation underlying this paper is the question of whether and how networks of neurons in the brain store *content*, as part of a larger thrust to explore the notion of representation in neural networks. I will focus on feedforward, classificatory neural networks (Rumelhart et al., 1986), for their relative ease of access together with their non-trivial computational potential (Sejnowski & Rosenberg, 1987; Churchland & Sejnowski, 1990). There are two layers of analysis I will implement. The first is part of the drive to understand the functioning of feedforward neural networks by analyzing their hidden layer state spaces (see Churchland (1989), for example). These abstract spaces, whose physical dimensionality is determined by the total number of units in the hidden layer, encode distributed activity across the hidden layer of a network. Within these spaces, I propose to identify geometrical structures called *polytopes* as vehicles of content, an assumption that follows naturally from the manner in which these networks carry out their tasks. The second layer of analysis places

*Preprint of a paper forthcoming in *Philosophical Psychology*. Date of original preprint: 30/9/2014. Current address: Department of History and Philosophy of Science, University of Cambridge, Free School Lane, Cambridge, CB2 3RH, U.K. Email: feraz.azhar@alumni.physics.ucsb.edu

the identification of polytopes on a firmer, quantitative footing, by utilizing Shannon's theory of information (Shannon, 1948; Shannon & Weaver, 1949). In particular, applying a novel definition of *informational content* that has its basis in the concept of mutual information, will allow us to identify their content in a principled manner. I utilize this analysis to help explicate the nature of misclassification in the neural networks considered, and present a new information-theoretic solution to the disjunction problem. In this way, this paper provides impetus to the assertion that content-bearing structures in networks of neurons partake of a causal-informational relationship with their surroundings (Usher, 2001).

The structure of this paper is as follows. In section 2, I discuss the broader context for the problem addressed herein. Section 3 contains a statement of the proposal, including the pertinent definition of informational content. I deploy this proposal in a suite of experiments that draw on both artificial and real data in section 4, to illustrate how these ideas work in practice. I argue that misclassification is naturally built-in to the analysis, and provide a novel way to think about the disjunction problem, introducing a new solution to the problem in section 5. In section 6, I consolidate these findings and point to future work that aims to establish the validity of the proposal in more general settings.

2. Localizing Content in Networks, Naturally

In attempting to understand states of the brain that are involved in higher-level conscious processes, a natural line of inquiry is to take a network-based, bottom-up point of view. These psychological states, such as beliefs, desires and propositional attitudes more generally, need not, of course, have simple explanations in terms of more basic networks. Indeed, there has been much controversy over the past 20 years or so, starting most notably with the work of Ramsey et al. (1990), in the context of exploring the utility of simple networks in understanding the psychological states that we refer to in our folk notions of psychology. Their contention was that for particular types of networks, there exists a difficulty in establishing a correspondence between the 'modularity' of psychological states and the states that networks adopt. If networks are then thought to faithfully model relevant brain functions, the folk psychological states we believe are being processed by the brain *do not exist*—a thesis known as 'eliminativism about propositional attitudes'.

I contend that the attempt to explore a potential mapping from the taxonomy of psychological states onto what the brain does, requires a systematic and comprehensive understanding of states manipulated at the level of neural networks. It may well turn out that this mapping can't be made precise, and that these psychological states are then indeed ill-defined, or that networks do not constitute the right level of analysis to understand these states (or some combination of the two). In any case, there is an important utility in attempting to explore this connection from a 'network-first' approach.

The goal in this paper is to establish a set of basic ideas that a network-first approach might rest on in a larger program investigating psychological states and their brain-like counterparts. A natural place to begin is to investigate regularities in the manner in which trained neural networks handle their inputs. I argue they do so by processing contentful (mental) states, which I will ascribe to physical spaces within these networks. The specific definition of content adopted will become clearer as the argument develops (one can skip to Definition 1 in section 3.3 for a preview), though for now our intuitive understanding of this concept will suffice.

For the sake of computational simplicity, I restrict attention to feedforward, artificial neural networks involved in classifying inputs into predetermined classes (schematically shown in Figure 1). Within this context, there exists a range of proposals for what constitute contentful structures. Working our way from the bottom-up (of Figure 1 say), one might

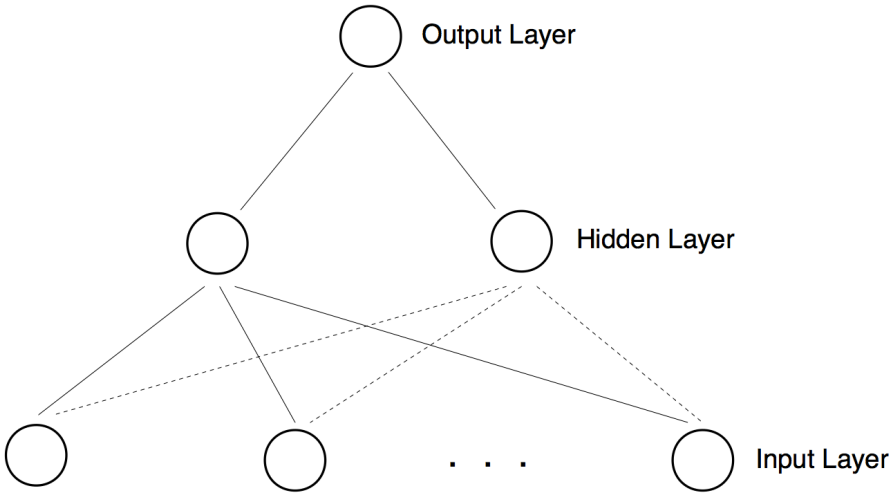

Figure 1: A hypothetical feedforward neural network with an unspecified number of input units, two hidden units and a single output unit. Solid or dashed lines represent potential connections from a unit in the preceding layer to a unit in the hidden or output layers.

contend that the weights connecting the input layer to the hidden layer house basic identifiable features of the inputs. O'Brien and Opie (2006) for example, argue that the weights that impinge upon any particular hidden unit, from the input layer, "*structurally resemble*" features of the inputs (O'Brien & Opie, 2006, p. 34). This claim is used as a basis for identifying these weights as forming a type of vehicle for representational content.

A second claim is that units in the hidden layer individually house content, in such a way as to provide a more complete picture of the input by combining individual contributions from each of the units. One reading of this possibility is that each unit corresponds to some feature relevant to differentiating inputs, and the degree of activation of any one unit represents the degree to which the input that causes this activation contains that feature (see Churchland (1998) and Shea (2007) for discussions of these types of claims).

A third, relatively more abstract way of placing content in these networks is to look into state spaces of their hidden layers. In this case, one might assert that individual activation points have different content, or argue for regions of points as being the structures that house content. This latter viewpoint corresponds most closely to that adopted in this paper, and finds its roots in work by Shea (2007), who holds that clusters in the state space of the hidden layer constitute vehicles of content, and by Churchland (1989, 2012), who (in one case) emphasizes the importance of partitions in the state space of the "crucial representational layer" (Churchland, 2012, pp. 7–9), in the context of Cottrell's face discrimination network (Cottrell, 1991).

To be clear then, this paper advocates analysis at the level of state spaces of the hidden layers of these networks, pointing to particular types of regions in these state spaces ('polytopes') as those that house content. The nature of these regions is defined by the need for the network to distinguish between inputs in a reliable way. A crucial feature of the analysis in this paper, and an important addition to the works mentioned above, is that it provides an independent, quantitative basis for identifying the content of these regions, which has its foundations in Shannon's theory of information (Shannon, 1948; Shannon & Weaver, 1949).

Shannon's theory was adapted to thinking about quantitative issues in the brain relatively soon after its introduction in 1948. Its uptake in addressing more conceptual issues was slower, perhaps because of its explicit lack of reliance on the *meaning* of messages conveyed during information transfer. Work on a semantic definition of information was instituted most prominently by Dretske in 1981 (Dretske, 1981; Lombardi, 2005). More recently, proposals inspired by the notion of mutual information, a symmetric quantity that expresses the average amount of information one variable conveys about another, have been invoked as a way to understand content in neurobiological systems (Eliasmith, 2000), as well as in the broader context of systems invoking conceptual representations, as introduced by Usher (2001).

This latter work is most pertinent for our discussion here. Usher argues for a statistical relationship based on mutual information, as the one that endows "primitive conceptual representations" or "concepts" with content corresponding to "objects" in the external world (Usher, 2001, p. 317). The idea builds upon the intuition that one can think of the content of a concept as referring to a class of objects in the world that is "*most likely*" to have caused the tokening of the concept (Usher, 2001, p. 316). On this reading, there is not only a causal connection between the object in the world and the concept, but a probabilistic one, which allows one to uniquely identify the content of a concept, in the face of the possibility that many objects may have caused it.

This paper adapts this work to a more restricted setting, addressing the challenge of developing a causal-informational framework for content, in the context of understanding the functioning of a restricted class of neural networks. In the course of doing so, I explicate how misclassification is naturally built-in to the formalism and provide a novel, information-theoretic solution to the disjunction problem in a particular setting. This latter problem, introduced by Fodor, reflects a commonly accepted stumbling block for theories that posit a correlational or causal basis for content-determination (Fodor, 1984; Usher, 2001; Shea, 2013).

3. Polytopes and Informational Content

I begin by developing the proposal for informational content, starting with a discussion of pertinent features of a stylized network that form the basis of the proposal, before turning to experimental considerations that will serve to further elucidate the idea.

3.1 Decision Regions for Classificatory Networks

Consider a hypothetical feedforward network with an input layer, 2 hidden units, and 1 output unit (as depicted in Figure 1). One can represent the computation carried out by the output unit by a nonlinear transformation f, acting on the postsynaptic potential impinging on the output unit, with

$$y = f(w_1 h_1 + w_2 h_2 + b), \qquad (1)$$

where y is the output activation level and f is sigmoidal, ranging from -1 to 1 in a continuous fashion (say). The argument of f is the postsynaptic potential, where $\{w_1, w_2\}$ are weights from the first and second hidden units to the output unit respectively, $\{h_1, h_2\}$ are respective hidden unit activation levels, and b is the output bias. I'll assume for the sake of simplicity that both hidden units also utilize the same nonlinearity f in processing their inputs.

Assume that this network has been trained on a set of input patterns to distinguish between two classes into which the inputs fall, namely, x_1 and x_2. The presentation of any one input pattern at the input layer stimulates the network, producing an activation level at the output, determined by Equation (1). The classificatory nature of this network is entwined with its ability to map exemplars of the two different input classes to reliably distinguishable output values. In principle, as is the case in supervised learning, we may have trained the network to map exemplars of input class x_1 to -1 and exemplars of input class x_2 to 1. In general we may not succeed in doing so exactly (even with the training set), but given the right circumstances, we might get close, in which case simple thresholds over the output unit's activation values will be enough to classify input patterns as either belonging to x_1 or x_2. It may well be that we cannot find thresholds that distinguish between every exemplar of the two classes perfectly, but having found ones that separate most of them, we might be willing to think of the network as having 'successfully' learnt to perform the task of classification. This latter case is perhaps the most realistic one, when thinking of these networks as proxies for what may be happening in highly reduced representations of real networks of biological neurons. If we think of this toy network as embedded in a much larger network that relies on information from the toy network to guide further action, a first-order solution for figuring out which input class was presented to the (trained) network, is indeed to just threshold over the output unit's activation level. It is the simplest approach to take, and the one that will be assumed here.

For these reasons, I will work within the paradigm where it is assumed a 'decision' is made by this toy network to classify an input pattern as belonging to x_1 say, when y takes a value below some critical value, say T_1. Similarly, I assume the network classifies the input as belonging to x_2, when y takes a value greater than or equal to some critical value T_2 (where $T_1 \leq T_2$). In other words, from the point of view of the classificatory nature of the network, it makes sense for the following interpretation to be afforded to the occurrence of any particular output level:

$$y = f(w_1 h_1 + w_2 h_2 + b) < T_1 \Rightarrow \text{Input belongs to } x_1, \qquad (2)$$

$$y = f(w_1 h_1 + w_2 h_2 + b) \geq T_2 \Rightarrow \text{Input belongs to } x_2. \qquad (3)$$

Note that if there is strict inequality between these two thresholds, i.e., $T_1 < T_2$, we might imagine that the network will treat the situation where $T_1 \leq y < T_2$ as indicating that the input belongs to neither class. In this paper, for the sake of simplicity, I will set $T_1 = T_2$.

3.2 Content in the Hidden Layer State Space

The decision made by this network can be represented diagrammatically, as the result of processing by the network at the state space of the hidden layer. In this case, the state space is two-dimensional, with points being labelled by activation levels of the two hidden units $\{h_1, h_2\}$. The two conditions given by Equations (2) and (3), partition the state space into 'decision regions', through the introduction of hyperplanes (which are simply lines in this two-dimensional example), determined by the values of the thresholds T_1 and T_2. The state space is thereby segmented into *polytopes*, i.e., regions demarcated by a finite number of hyperplanes, that map each pair of activation levels $\{h_1, h_2\}$, into one of two categories, i.e., x_1 or x_2. An illustrative depiction of this scenario is presented in Figure 2.

A key point is that these decision regions are a generic feature of the state space of the hidden layer in feedforward classificatory neural networks. For the network introduced in section 3.1, *each point* in the state space will get mapped to a single output value y through the transformation in Equation (1). Depending on what that value is, the output unit will be

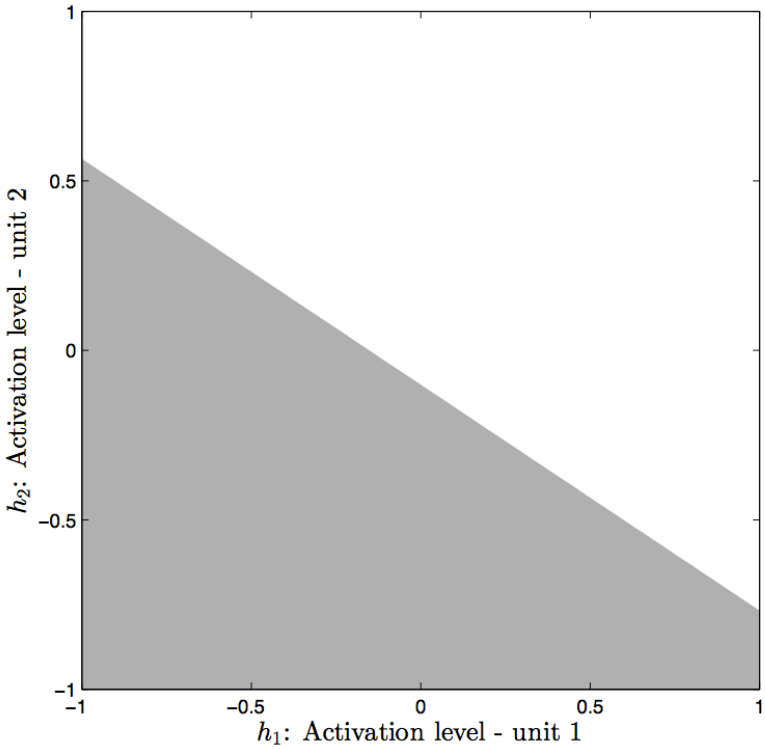

Figure 2: State space of the hidden layer for the toy network introduced in section 3.1. The horizontal axis displays the activation level of the first hidden unit, h_1, with the vertical axis displaying the corresponding value for the second unit, h_2. Thresholds that define the classification scheme (in Equations (2) and (3)) are assumed to be zero, i.e., $T_1 = 0 = T_2$. The white polytope corresponds to the area that gets interpreted as having been activated by an exemplar of input class x_2, with the grey polytope being interpreted as having been activated by x_1. Parameters used to produce the plot were chosen arbitrarily, with $w_1 = 2$, $w_2 = 3$ and $b = 0.3$ (see Equation (1)). The nonlinearity was set to be $f(x) = \tanh(x)$.

able to transmit information to units or networks downstream, indicating the fact of the earlier occurrence of an input of class x_1 or x_2, according to the classification scheme the output unit effectively implements (Equations (2) and (3)).

This structure is generalized in a straightforward way to systems with more than one output unit. Multiple output units would tile the hidden layer state space with overlapping sets of polytopes that segment the state space into distinct regions, each of which would also constitute a polytope. Activation of any particular region would correspond to activation of the output units in a particular way, and would thereby signal the occurrence in the external environment, of whatever that network was trained to recognize with those levels of activation (through an appropriate generalization of Equations (2) and (3)). This paper will focus on the case of a single output unit for the sake of simplicity.

In a sense, these decision regions are clusters of points in the state space of the hidden layer that reliably signal the occurrence of a specific class of input in the network. One can think of them as having *content* about the decision they are constructed to help make—namely, that *the input belonged to a specific class*. This interpretation is not particularly controversial, given that this is effectively the task the network is designed to do. We can put this interpretation on a firmer footing however, particularly given that classification accuracy

will not necessarily be perfect for any given network. In the following section, I introduce a more robust definition of the content of these polytopes that invokes the theory of information as first outlined by Shannon (1948).

3.3 Polytopes as Vehicles of Informational Content

Though the scheme implicit in Equations (2) and (3) seems intuitive, a measure for ascribing content to any region in the state space of the hidden layer that is insensitive to small deviations from perfect (linear) separability of the input classes, seems necessary for applying this framework to more general settings. In other words, we are looking for a principled and quantitative way to unambiguously specify the content of each polytope, given that it has been trained to distinguish input classes in the manner assumed.

The proposal here builds on the work of Usher (2001), who develops a statistical scheme for content determination, for a generic system that represents objects in the external world through the tokening of an unspecified representational vehicle. I focus a modified version of this account on the manner in which polytopes may well be used by networks to identify specific classes of input, thereby endowing them with what might be termed 'informational content'.

Consider a system where there are M polytopes in the trained state space of the hidden layer of a feedforward neural network, and N possible classes of inputs where $N \leq M$. Let $X = x_1, x_2, \ldots, x_N$ be the random variable representing the inputs, where x_n represents input class n. Let $Y = y_1, y_2, \ldots, y_M$ be the random variable representing the polytopes, where y_m represents (the activation of) polytope m. A natural measure of association between the two random variables is the mutual information:

$$I(X;Y) = \sum_{n=1}^{N} \sum_{m=1}^{M} p(x_n, y_m) \log_2 \frac{p(x_n, y_m)}{p_X(x_n) p_Y(y_m)}, \tag{4}$$

which expresses the average amount of information that Y produces about X (Cover and Thomas, 1991). The joint probability $p(x_n, y_m)$, is the one that is read off from the experiment in question, with the marginals, $p_X(\cdot)$ and $p_Y(\cdot)$, defined in the usual way as sums over this joint distribution, e.g., $p_X(x_n) = \sum_{m=1}^{M} p(x_n, y_m)$. One can rewrite this in a slightly more illuminating way by noticing that

$$\begin{aligned} I(X;Y) &= \sum_{n=1}^{N} \sum_{m=1}^{M} p(x_n, y_m) \log_2 \frac{p(x_n|y_m)}{p_X(x_n)} \\ &= \sum_{n=1}^{N} \sum_{m=1}^{M} p(x_n, y_m) \left[\log_2 \frac{1}{p_X(x_n)} - \log_2 \frac{1}{p(x_n|y_m)} \right]. \end{aligned} \tag{5}$$

The term in square brackets on the right hand side of Equation (5), sometimes referred to as the pointwise mutual information, $i(x_n, y_m) := \log_2(1/p_X(x_n)) - \log_2(1/p(x_n|y_m))$, has an elegant information-theoretic interpretation. It is the reduction in our surprise about the occurrence of input class x_n, given that polytope y_m was activated. In other words, one can think of it as the information that polytope y_m carries about the input class x_n. It is clear that in the case one is interested in identifying a polytope with a particular input class, one would want to restrict attention to those input classes that satisfy $i(x_n, y_m) > 0$ for any polytope y_m. In principle, there may exist many input classes for any particular polytope that have this

property, and one can use the joint probability pre-factor $p(x_n, y_m)$, which multiplies $i(x_n, y_m)$ in the definition of the mutual information to (hopefully) single out a particular input class as the one that the polytope carries the most (weighted) information about. From these considerations, I propose the following definition for identifying the informational content of a polytope:

Definition 1. For any polytope y_m, its **informational content** is 'input class x_n obtained in the network' (or just 'x_n' for short), where this input class is the one that has the maximal contribution to the mutual information, compared to all other input classes, subject to the constraint that the contribution is positive. That is, for any $m \in \{1,2,\ldots,M\}$, the polytope represented by y_m has the informational content 'x_n', where x_n is the input class that both maximizes

$$\mathcal{M}(x_n, y_m) := p(x_n, y_m) \log_2 \frac{p(x_n, y_m)}{p_X(x_n) p_Y(y_m)}, \tag{6}$$

and satisfies the constraint that

$$\mathcal{M}(x_n, y_m) > 0. \tag{7}$$

Demanding $\mathcal{M}(x_n, y_m) > 0$ enforces the earlier desideratum that $i(x_n, y_m) > 0$. For the sake of clarity, we'll investigate how this definition works in a simplified context before applying it to more realistic situations in section 4.

Consider the case where the network of section 3.1 has been trained to distinguish two input classes, x_1 and x_2, yielding a two-dimensional hidden layer state space in two separate instances (starting with different initial weight configurations for example), as shown in Figure 3. Assume the thresholds that define the classification scheme (in Equations (2) and (3)) are equal to zero, i.e., $T_1 = 0 = T_2$. The white polytopes in Figure 3 are labelled by y_2 and the grey ones by y_1, and assume that the network was trained to map exemplars of x_1 to a value represented by y_1, with exemplars of x_2 mapped to y_2. Activation points that result from presenting each input pattern to the trained network have been superimposed, with those corresponding to exemplars from input class x_1 shown as red triangles, and those of input class x_2 shown as green circles. The total number of input samples used to train the network was assumed to be 10 (for the sake of simplicity), though the precise number is not important for our considerations.

In Figure 3a, the polytopes cleanly separate the two categories, while in Figure 3b, they do not. The state space in Figure 3b misclassifies one of the exemplars belonging to x_2 as belonging to x_1. To apply our definition of informational content to this situation, one begins by constructing the matrix of joint probabilities over the input classes $\{x_1, x_2\}$ and the polytopes $\{y_1, y_2\}$. We'll focus solely on the more interesting case of Figure 3b. There one finds

$p(x_n, y_m)$	y_1	y_2
x_1	0.4	0
x_2	0.1	0.5

From Equation (6), $\mathcal{M}(x_n, y_m)$ is then

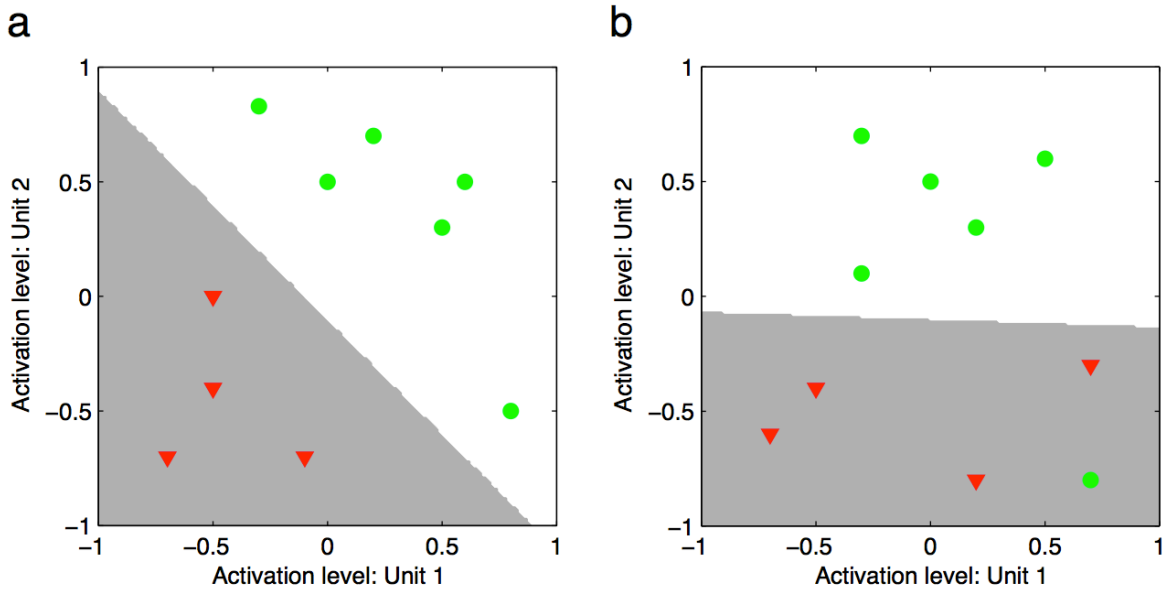

Figure 3: Two state spaces of the hidden layer for the hypothetical model introduced in section 3.1. By assumption, the network has learned two different sets of values for the weights, resulting in polytopes with different sizes and orientations. The green circles are activation points generated from inputs belonging to x_2, while the red triangles are activation points belonging to x_1. The state space on the right (b), is evidently mislabelling one of the x_2 inputs. All parameters have been chosen arbitrarily for the sake of illustration.

$\mathcal{M}(x_n, y_m)$	y_1	y_2
x_1	0.400	0
x_2	-0.158	0.368

Applying Definition 1 to this matrix gives that the grey polytope (y_1) has the content that the red triangles (x_1) occurred in the network, while the white polytope (y_2) has the content that the green circles (x_2) occurred in the network. Both of these conclusions accord with what we expect intuitively. Evidently, given that the output unit in the network will project the activation of any point in the grey polytope onto the presumption of the occurrence of a red triangle, the single green circle residing in the grey polytope of Figure 3b will be misclassified—an issue we will explore in more depth in section 5.

4. Content in Experimental Neural Networks

To show how the definition of informational content introduced in section 3.3 applies in more concrete settings, I will investigate three distinct sets of networks. The first set involves an analysis of synthetic data, where networks are trained to recognize the result of a simple transformation of the inputs. In the second set, sonar signals obtained from reflection off one of two different media are classified. In the final set, I'll analyze data dealing with classification of Single-Photon Emission Computed Tomography (SPECT) images from patients being monitored for cardiac abnormalities, creating networks designed to classify images that are usually done so by hand. I'll begin by describing the experiments, displaying

representative polytopes of trained hidden layer state spaces, before deploying Definition 1 to unambiguously identify the informational content of each region in section 4.2.

4.1 Experimental Classificatory Networks

At the outset, a few technical details common to all networks constructed are in order. Simulations were performed in MATLAB (R2013a: Student Version. The MathWorks, Inc., Natick, MA, USA), utilizing scaled conjugate gradient backpropagation as part of the Neural Network Toolbox. For each network, a single output unit was utilized. The thresholds that define the resulting polytopes are set as in the example of Figure 2, with $T_1 = 0 = T_2$. The nonlinearity used for each neuron in the hidden and output layers was $f(x) = \tanh(x)$ (instituted throughout this paper using the 'tansig' function). In each case, the entirety of the data was used to train the network. Once training stopped (according to a set of preassigned conditions), a network was judged to have successfully learnt the task if the area under the receiver operator characteristics curve, comparing the targets with the output of the trained network on each input pattern, was greater than or equal to 90%.

4.1.1 Uniformly sampled input sums

In this first example, feedforward neural networks were trained to execute a simple categorization task that required each network to recognize the sum of the inputs. The total number of input units ranged over $N_{\text{input}} = 5,6,7,8,9$, with the number of hidden units set to $N_{\text{hidden}} = 2,3, \text{or } 4$. A total of 15 distinct network architectures were thereby constructed, which will be labelled by the pair $\{N_{\text{input}}, N_{\text{hidden}}\}$.

Inputs were obtained by (pseudo)randomly sampling from a uniform distribution over the interval $[-1, 1]$. If the sum of the inputs over the input layer was greater than or equal to 0, the input was mapped to 1, otherwise it was mapped to -1. The total number of patterns the network was trained on for each value of N_{input} was chosen to be 200; the split between the two input classes was roughly even.

In Figure 4, the state space of the hidden layer after training is displayed, for the case $\{N_{\text{input}} = 5, N_{\text{hidden}} = 2\}$. The figure displays the state space for each of 20 separate random initializations of the weights and biases for the network. Activation patterns as generated by the two classes of inputs fed into the network are also superimposed, showing perfect separation of the two classes in each case.

In Figure 5, the corresponding plot for the case where the number of hidden units is $N_{\text{hidden}} = 3$ is shown. Perfect separation of the input patterns occurs, except for a single network, which is evidently misclassifying exemplars (see caption). Polytopes are capable of *misclassifying* the input class, a phenomenon that will become more apparent in the two types of networks presented in sections 4.1.2 and 4.1.3. This will have important implications for the viability of our proposal.

Results for the other networks with $N_{\text{hidden}} = 2,3$ were similar to those displayed here, with networks also successfully learning to complete the task for $N_{\text{hidden}} = 4$.

4.1.2 Sonar echoes

The second class of example networks attempts to target a more realistic scenario, as described by experiments developed by Gorman and Sejnowski (1988). They were interested in classifying sonar echoes resulting from reflection off undersea targets falling into one of two categories, either rock or metal. They present a dataset containing 208 sonar echo samples (111 metal returns, 97 rock returns), each with 60 dimensions encoding the energy

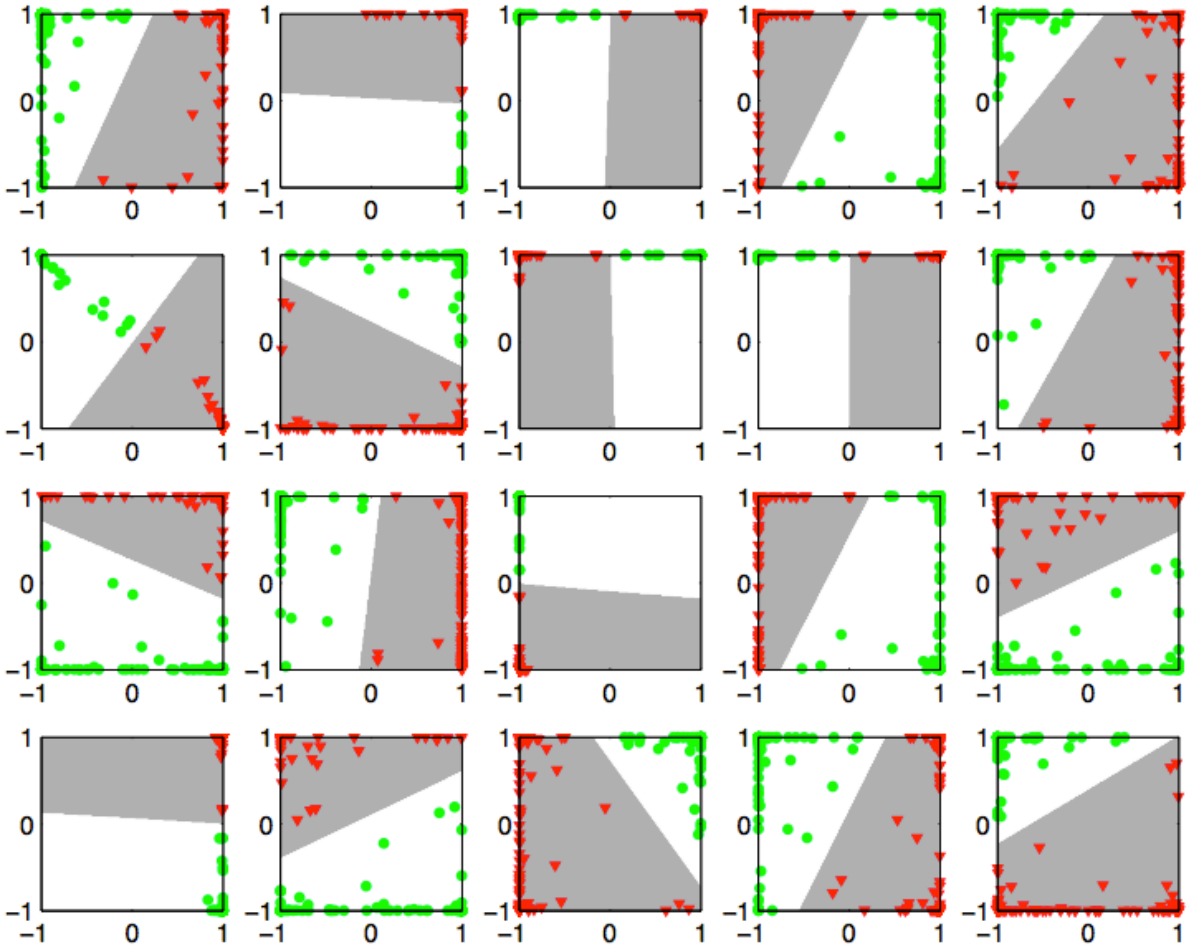

Figure 4: State space of the hidden layer for the uniformly sampled input sum network $\{N_{\text{input}} = 5, N_{\text{hidden}} = 2\}$. Results from 20 separate networks are displayed, where the initial weights and biases were randomized for each network. The white polytope corresponds to the region denoting the sum over the inputs was greater than or equal to zero (a target of 1), with the grey polytope corresponding to the region denoting a sum of less than zero (target of -1). The green circles correspond to those inputs with a sum of greater than or equal to zero, and the red triangles correspond to those inputs with a sum of less than zero. Perfect separation of the input classes occurs in each instance, in agreement with the requisite classification task.

measured within some frequency band, integrated over a particular time period (data was downloaded from the UCI Machine Learning Repository (Bache & Lichman, 2013)). The different returns encoded a variety of different aspect angles invoked during the experiment for each target (see Gorman and Sejnowski (1988) for further information).

Networks were trained to distinguish sonar echoes from the two different media. The total number of input units was set at $N_{\text{input}} = 60$, corresponding to the 60 data points produced for each echo, and a total of $N_{\text{hidden}} = 2,3,$ or 4 hidden units was used.

Figures 6 and 7 display the two- and three-dimensional state spaces of the hidden layer as obtained after the networks were trained. Note that although the networks learned to classify the two categories to the requisite accuracy level, the polytopes in the state space of

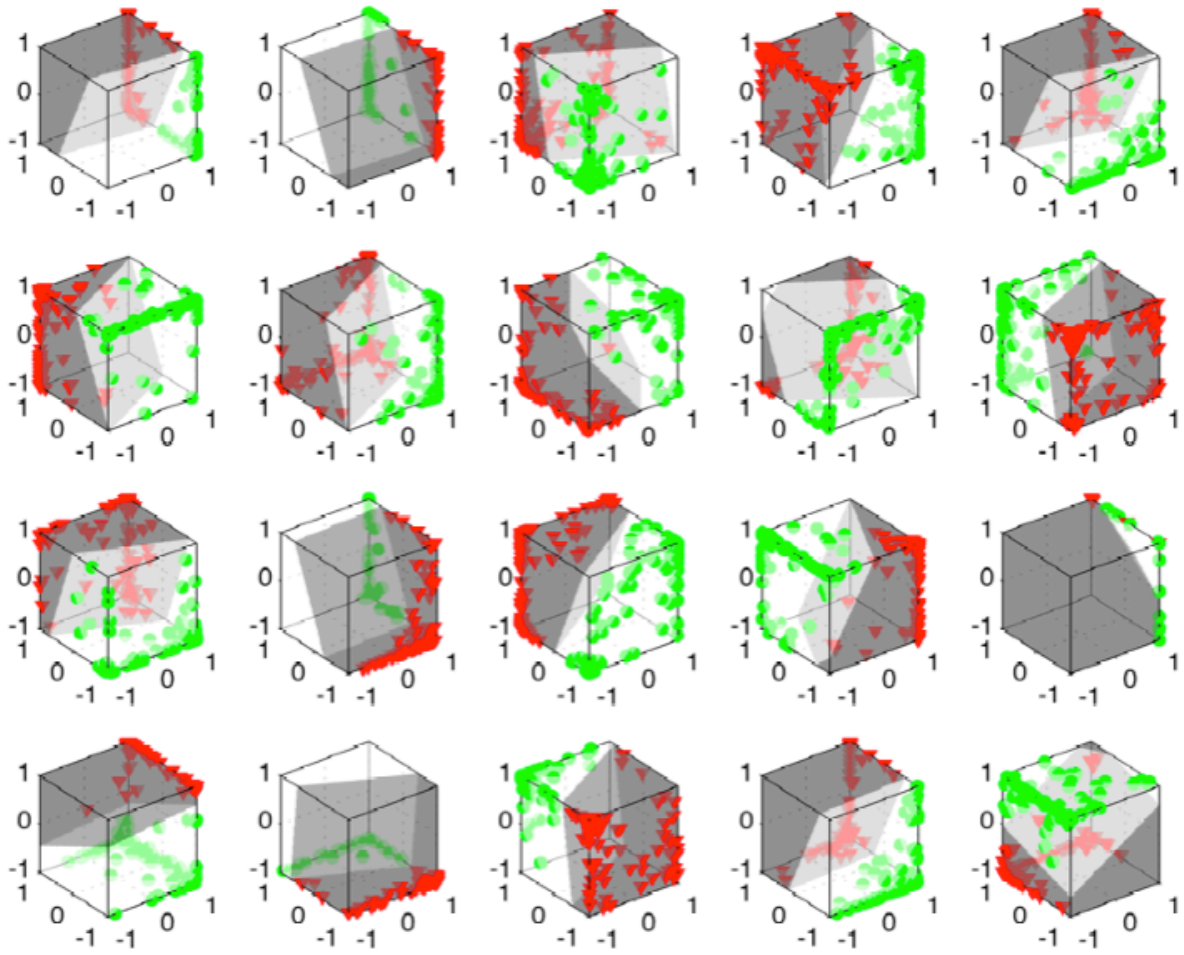

Figure 5: State space of the hidden layer for the uniformly sampled input sum network $\{N_{\text{input}} = 5, N_{\text{hidden}} = 3\}$. Results from 20 separate networks are displayed, where the initial weights and biases were randomized for each network. The color scheme is as described in the caption to Figure 4. Perfect separation of the input patterns for each network occurs, save for one (third row, fifth column).

the hidden layer do not generally segment the state space into mutually disjoint classes. In the case of $N_{\text{hidden}} = 4$, networks also learned to successfully classify the two categories of input.

4.1.3 Cardiac SPECT imaging

This final class of networks is involved in the classification of cardiac SPECT images as analyzed by Kurgan et al. (2001). They produced a dataset of images obtained from 267 patients as they were undergoing monitoring during myocardial perfusion studies (data was downloaded from the UCI Machine Learning Repository (Bache & Lichman, 2013)). For each patient's imaging study, a total of 44 features were extracted, corresponding to the degree of perfusion inside a particular region of interest during one of two conditions, when the patient was either under stress or at rest. Of these images, 212 were 'abnormal', while 55 were 'normal' (see Kurgan et al. (2001) for further details).

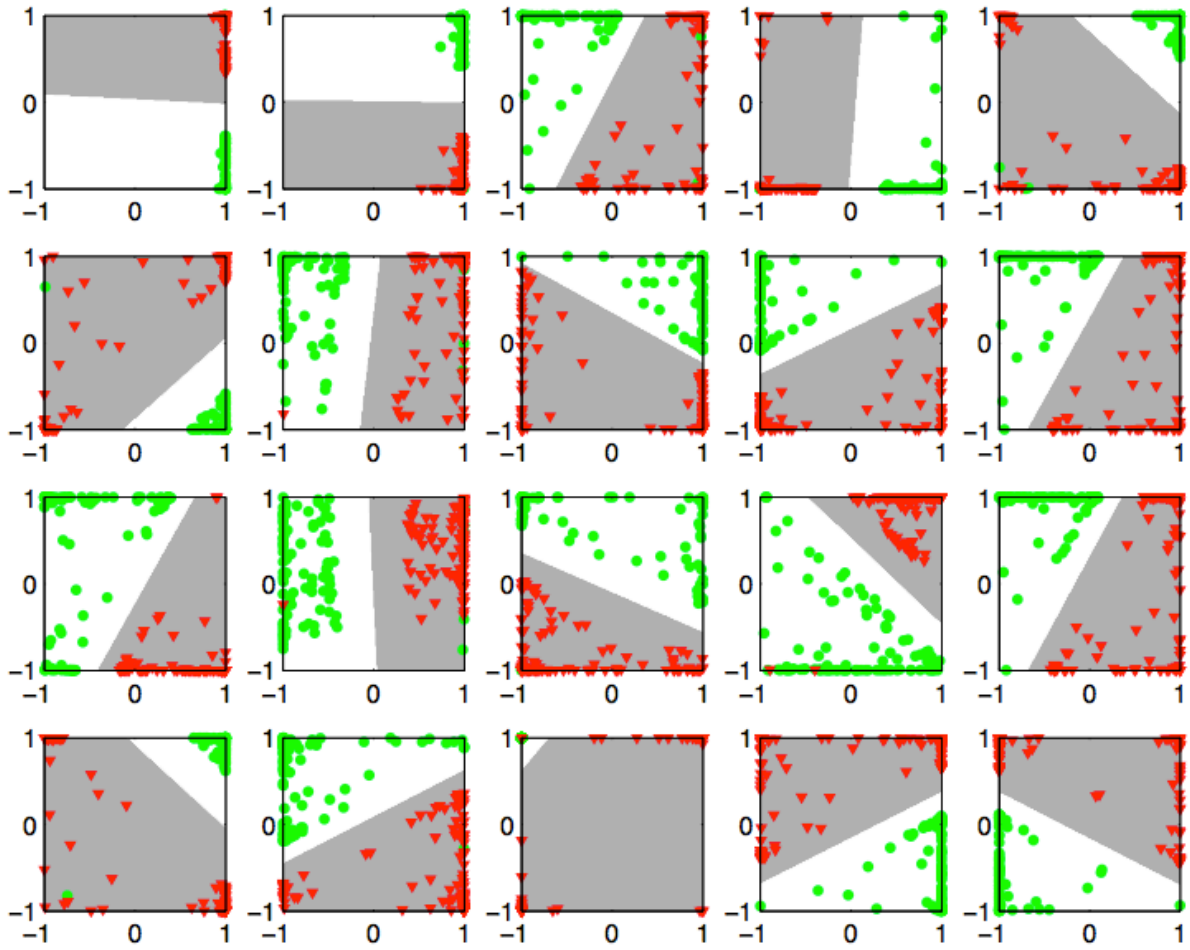

Figure 6: State space of the hidden layer for the sonar classification network $\{N_{\text{input}} = 60, N_{\text{hidden}} = 2\}$. Results from 20 separate networks are displayed, where the initial weights and biases were randomized for each network. The white polytope corresponds to the region denoting the existence of rock on the seabed (a target of 1), with the grey polytope corresponding to the region denoting the existence of metal (target of -1). The green circles correspond to those inputs that were indeed a result of reflection off rock, whereas the red triangles correspond to those inputs that encoded reflection off metal. Note the existence of misclassifications for some of these networks.

Networks were trained to distinguish between abnormal and normal images. The total number of input units was set at $N_{\text{input}} = 44$, corresponding to the 44 features produced for each patient's study, and a total of $N_{\text{hidden}} = 2,3,$ or 4 was used.

Figures 8 and 9 display the two- and three-dimensional state spaces of the hidden layer as obtained after the networks were trained. As in the case of the sonar classification task, the networks learned to classify the two categories to the requisite accuracy level, though here the polytopes in the state space of the hidden layer do not segment the state space into mutually disjoint classes for any network. Again in the case of $N_{\text{hidden}} = 4$, the network learned to successfully classify the two input categories.

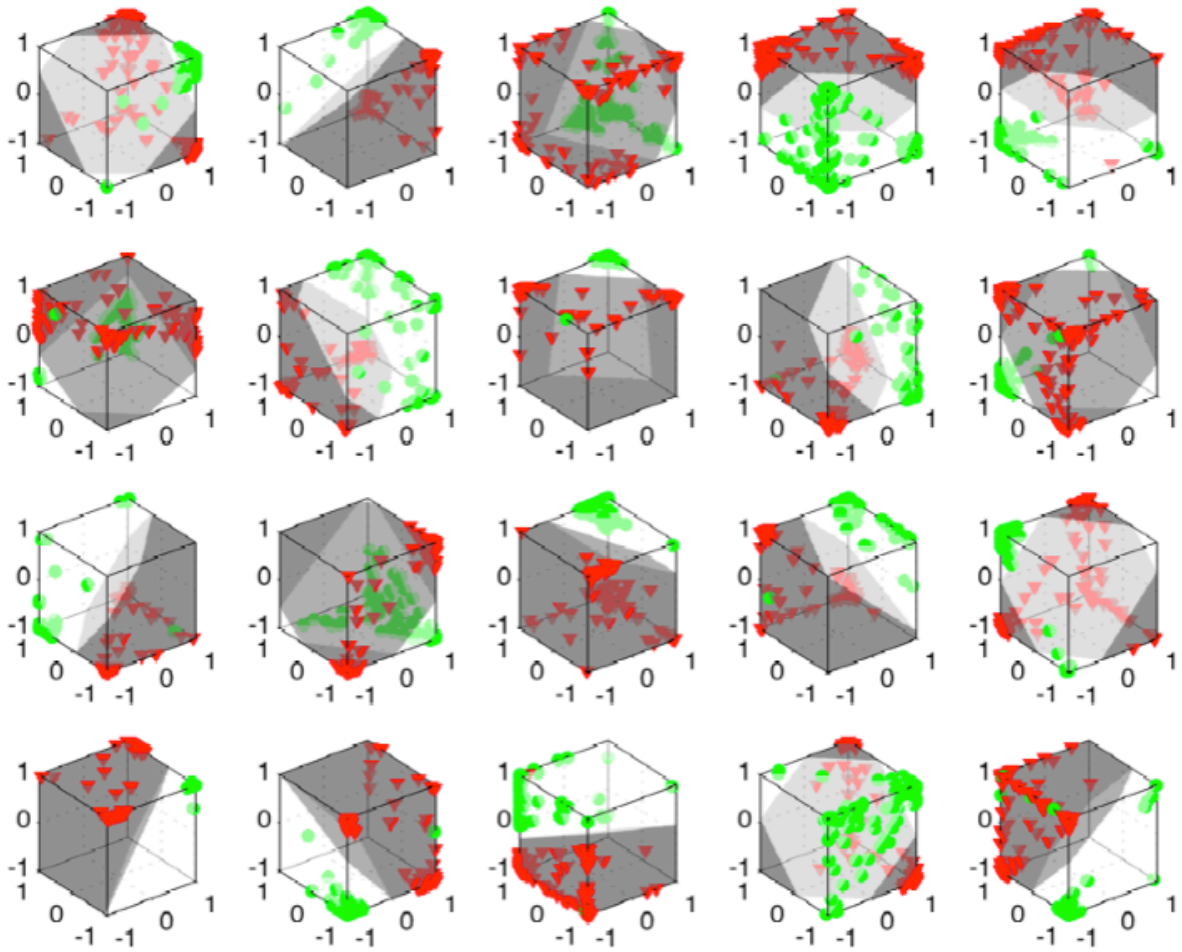

Figure 7: State space of the hidden layer for the sonar classification network $\{N_{\text{input}} = 60, N_{\text{hidden}} = 3\}$. Results from 20 separate networks are displayed, where the initial weights and biases were randomized for each network. The color scheme is as described in the caption to Figure 6. Note the existence of misclassifications for some of these networks.

4.2 Analysis of Informational Content

The results from section 4.1 that show clean separation of input classes at the level of the state space of the hidden layer, trivially satisfy the conditions specified in Definition 1 for determining informational content. The cases that are most interesting for our purposes here (and in what follows), correspond to the situation where despite the network having been judged to have attained the requisite level of accuracy, the two input classes are not cleanly separated. This occurs predominantly in the cases corresponding to the sonar return classification task and the cardiac SPECT image classification task.

In this section, I take examples from the three network types previously considered, and show how the above prescription for informational content is consistent with what one might expect intuitively, given the tasks the networks were designed to perform. At the outset, note that in every network instantiation considered in this paper, corresponding to 420 networks in total (including multiplicities derived from the random weight and bias

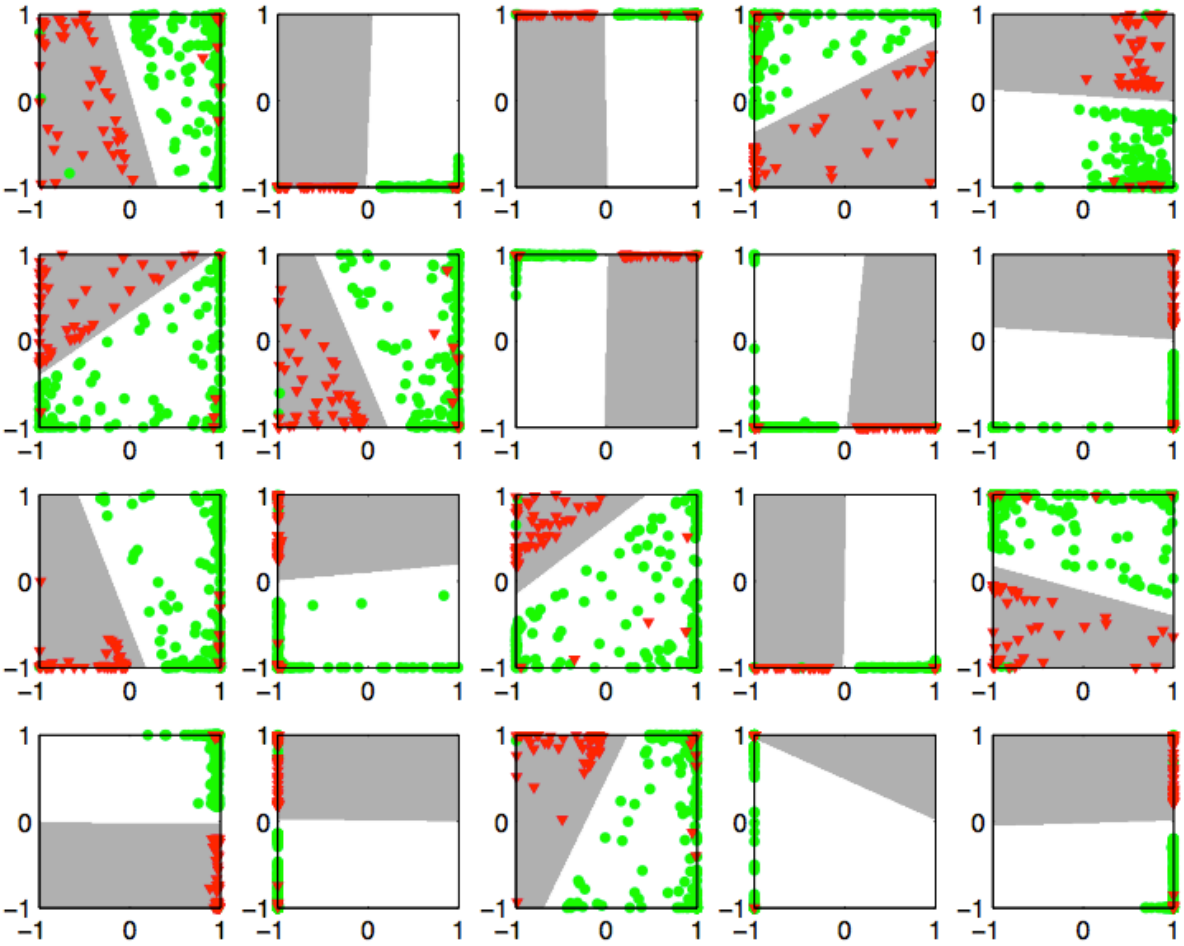

Figure 8: State space of the hidden layer for the cardiac SPECT classification network $\{N_{\text{input}} = 44, N_{\text{hidden}} = 2\}$. Results from 20 separate networks are displayed, where the initial weights and biases were randomized for each network. The white polytope corresponds to the region denoting an abnormal cardiac SPECT image (a target of 1), with the grey polytope corresponding to the region denoting a normal image (target of -1). The green circles correspond to those inputs that were indeed abnormal, with the red triangles corresponding to those inputs that were normal. Note the existence of misclassifications for each network.

initializations for each architecture), the prescription uniquely specifies the content of each polytope in agreement with the task domain.

Consider first the set of networks trained to recognize the sum of the inputs, presented in section 4.1.1. In particular, in the case of Figure 4, each network cleanly separates the two input classes and so the analysis of informational content is equivalent for all 20 networks. For any network then, one can show that the joint probability distribution, and the weighted pointwise mutual information $\mathcal{M}(x_n, y_m)$, which appears in Definition 1, are given by

$p(x_n, y_m)$	y_1	y_2
x_1	0.490	0
x_2	0	0.510

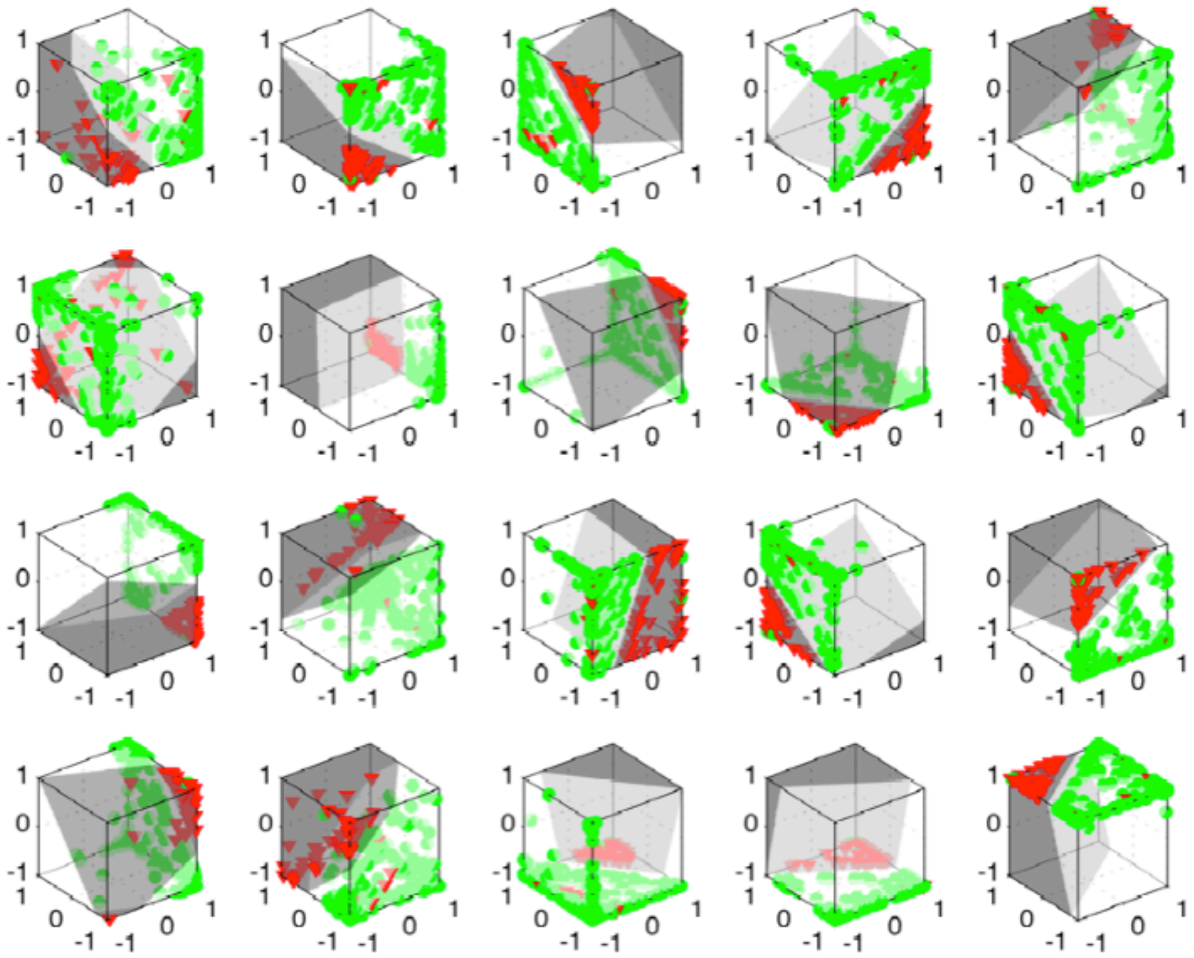

Figure 9: State space of the hidden layer for the cardiac SPECT classification network $\{N_{\text{input}} = 44, N_{\text{hidden}} = 3\}$. Results from 20 separate networks are displayed, where the initial weights and biases were randomized for each network. The color scheme is as described in the caption to Figure 8. Note the existence of misclassifications for each network.

$\mathcal{M}(x_n, y_m)$	y_1	y_2
x_1	0.504	0
x_2	0	0.495

where in an obvious notation, $\{x_1, x_2\} = \{\text{sum} < 0, \text{sum} \geq 0\} = \{\text{red triangles, green circles}\}$, and $\{y_1, y_2\} = \{\text{grey polytope, white polytope}\}$. In this case, Definition 1 implies that the content of a grey polytope is that the input pattern had a sum of less than 0, whereas the content of a white polytope is that the input pattern had a sum of greater than or equal to 0.

For the sonar classification task, I'll perform a similar analysis but for a single network appearing in the second row and second column of the two-dimensional state spaces of Figure 6. There one finds

$p(x_n, y_m)$	y_1	y_2
x_1	0.529	0.005
x_2	0.024	0.442

$\mathcal{M}(x_n, y_m)$	y_1	y_2
x_1	0.445	-0.027
x_2	-0.082	0.480

where $\{x_1, x_2\} = \{\text{metal, rock}\} = \{\text{red triangles, green circles}\}$ and $\{y_1, y_2\} = \{\text{grey polytope, white polytope}\}$. In this case, in the presence of clear examples of misclassification by the network, Definition 1 implies that the content of the grey polytope is that the sonar target was metallic, whereas the content of the white polytope is that the sonar target was a rock.

For the cardiac SPECT classification task, I'll focus on a single three-dimensional example appearing in the fourth row and second column of Figure 9. There one finds

$p(x_n, y_m)$	y_1	y_2
x_1	0.184	0.022
x_2	0.004	0.790

$\mathcal{M}(x_n, y_m)$	y_1	y_2
x_1	0.413	-0.065
x_2	-0.020	0.231

where $\{x_1, x_2\} = \{\text{normal, abnormal}\} = \{\text{red triangles, green circles}\}$ and $\{y_1, y_2\} = \{\text{grey polytope, white polytope}\}$. Definition 1 implies that the content of the grey polytope is that the SPECT study presented was normal, whereas the content of the white polytope is that the SPECT study presented was abnormal.

For the successfully trained networks displayed above then, the definition for content as determined by the information-theoretic measure in Definition 1 yields results in accord with what one might demand from such a definition, given the tasks these networks were designed for. In what follows, I'll delve into conceptual advances that these results support, turning first to the issue of misclassification.

5. Misclassification and a Novel Solution to the Disjunction Problem

The information-based analysis of content advocated in this paper picks a single input class for each polytope, yet a polytope can have more than one input class mapped to it. I claim that the activation of a polytope by an input class that is not part of its informational content is an example of misclassification. That is, the polytope signals the occurrence of a class of input that was not the one that triggered the polytope in the first place.

How this might come about operationally is related to the manner in which polytopes represent regions of similarity and difference with respect to the task domain. Once the weights and biases of the network have been fixed through training, similar inputs will get mapped to nearby regions in the state space of the hidden layer (in the absence of noise). A natural explanation for the origin of this misclassification is that there are certain similarities between the input class the polytope is tied to (by virtue of our information-theoretic definition), and a misclassified exemplar of an input class the polytope is not tied to, *that cannot be distinguished by the network*. It may well be that a more refined network will be able to distinguish the two classes more cleanly, but within the constraints of the artificial

network (a situation that mimics the existence of biological constraints on real networks of neurons), there is overlap between the input classes at the level of the state space.

I do not mean to argue that one can do away with misclassification entirely if only one knows which network to employ (and that therefore misclassification doesn't or shouldn't exist in this framework), the idea is that one can account for particular examples of misclassification in the context of the restricted computational power of the networks considered here, and that given the task of *classification* that any such network is designed to perform, misclassification is not necessarily excluded as a possibility in principle. This fact is consistent with Definition 1. Misclassification is possible in the context of this definition because the content of a polytope isn't tied to precisely which input class caused its tokening—rather, content is defined through a statistical relation that picks out the input class that contributes the greatest amount to the mutual information, out of all possible input classes, such that this contribution conveys a positive amount of information.

Another arm of the problem of accounting for misclassification as made explicit by Fodor (1984) (see also Usher (2001) and Shea (2013)) is: why choose a single input class as the one signalled by the occurrence of a polytope? In the case there exist two input classes, say x_1 and x_2, which both get mapped to the same polytope, why isn't the content of that polytope the disjunction $x_1 \vee x_2$? If this were the nature of the interpretation one applies to the networks created above, misclassification would be impossible, in that the activation of a polytope could not under any circumstance, indicate the occurrence of an input class that was not within the extent of its content. In Fodor's words:

> Here's the problem: R represents the state of affairs with which its tokens are causally correlated. Some representations of type R are causally correlated with states of affairs of type S; some representations of type R are causally correlated with states of affairs of type T. So it looks as though what R represents is not either S or T, but rather the disjunction $(S \vee T)$: The correlation of R with the disjunction is, after all, *better than* its correlation with either of the disjuncts and, ex hypothesi, correlation makes information and information makes representation. (Fodor, 1984, p. 240)

Though Fodor's discussion centers around a broader notion of representation than the one I am willing to commit to here, his explication of the *disjunction problem* has clear implications for the proposal in this paper.

Indeed, the information-theoretic structure introduced affords a novel solution to the disjunction problem. The activation of a polytope *does not* 'correlate' more strongly with the disjunction of the input classes. In fact, and this is true for whatever number of input classes one chooses, a polytope provides *zero information* about a strict disjunction over the entirety of the set of input classes. The reason for this is that the disjunction over the input classes *always occurs*. One learns nothing about which input class obtained in the network given the tokening of any particular polytope, because one already knows that *an* input class obtained. In the terminology introduced above, the probability of $x_1 \vee x_2$ is 1, and so there is nothing to be gained by monitoring whether or not any polytope was activated.

More formally, for any polytope y_1 say, form the disjunction over the inputs $x_{\text{or}} = x_1 \vee x_2$, for example. Then $p(x_{\text{or}}, y_1) = p_Y(y_1)$ and $p(x_{\text{or}}) = 1$; applying Definition 1 gives $\mathcal{M}(x_{\text{or}}, y_1) = 0$, and so y_1 does not have the disjunction as its informational content. This argument holds for however many input classes one wishes to build into the analysis.

It is not difficult to see that this solution extends beyond the bounds of the specific class of networks considered in this paper, encompassing coarser grained descriptions of input classes and content-bearing vehicles. In the context discussed by Usher for example,

when there exist multiple classes of external physical objects, say $\mathcal{S}_i$, that token a particular concept in the brain, say $\mathcal{R}$, the exhaustive disjunction $\bigvee_i \mathcal{S}_i$ will not be the informational content of $\mathcal{R}$ (under the appropriate analogue of Definition 1), because the occurrence of the disjunction is by definition, uninformative.

The crux of this solution rests on the fact that the nature of the disjunction problem relies heavily on how one specifies the statistical relationship between an input class and a vehicle of content. The quantitative measure of (probability-weighted) pointwise mutual information as a means of defining Fodor's notion of 'correlation' achieves two goals; the first is that it makes the disjunction problem well defined (i.e., refers to a precise notion of correlation), and the second is that it directly addresses the central assumption that underlies the existence of the disjunction problem, which in Fodor's words is "... ceteris paribus, if the correlation of a symbol with a disjunction is better than its correlation with either disjunct, it is the disjunction, rather than either disjunct, that the symbol represents" (Fodor, 1984, p. 240). Information as interpreted in Shannon's sense, and in particular, pointwise mutual information as utilized in Definition 1, ensures that the 'correlation' of a symbol with a disjunction won't be better than its correlation with either disjunct.[1]

6. Discussion

Establishing loci of content in neural networks provides a first step to constructing a more conceptual understanding of how these networks work. I have pointed to geometrical structures, which I've identified as polytopes in the state space of the hidden layer, as natural anchors for content, explicating the sense in which simple classificatory feedforward networks reference these structures. Some care needs to be taken however, in delimiting why one might want to refer to them in the first place, what relationship they bear to their constitutive (bio)physical structures, and why we might want to use mutual information as the measure underlying content determination—I address these questions in this section.

6.1 The Nature of Polytopes and their Content

Polytopes inherit their structure from the signalling mechanisms the output neuron in the network is engaged with. The correspondence between these artificial neural networks and real networks rests on the assumption that there is an inherent message contained in a particular activation level of some output unit say, and that sufficiently different activation levels signify different messages. Correspondingly, similar activation levels signify similar messages. Polytopes in the state space of the hidden layer segment the space into regions that activate the output unit in similar ways and can therefore be thought of as carrying similar messages.

To further our understanding of how these messages are localized in the networks studied above, it is important to explicate the nature of the relationship between polytopes and the (bio)physical structures they impinge upon. There exists, of course, a strict mapping between polytopes and the architecture of networks they exist within. Both the number and arrangement of neurons in the network, as well as the final weights and biases the network converges to during training, determine their physical characteristics. In particular, it is most natural to advert to these vehicles at the level of the hidden layer. The reason for this is that input layer encodings of input classes consist of unprocessed activation levels. There presumably exists information about the input class as encoded in the input activation patterns, but it is the job of the trained network to figure out what features of the encoding are relevant for the classification task. As a result, in principle, clustering of input patterns may not translate to clustering at the level of the output (see Laakso and Cottrell (2000) for a

quantitative illustration of this idea), and so asserting that input activation spaces are content bearing is difficult to justify. This leaves only the hidden layer, which as detailed above, inherits its structure from the decisions the output unit makes.

The physical dimensionality of our content-bearing vehicles is the same as the number of hidden units in the network, as the activation level of each hidden unit provides one dimension in the abstract space in which polytopes reside. The orientation and size of these vehicles are determined through training however, and depend, as we have seen qualitatively, on the initialized weights and biases of the network. This means that networks with the same architecture can possess content-bearing vehicles that signal the same thing, despite the precise values of their weights and biases being different.

One can take this analysis one step further and note that it is clear that different networks can solve the same problem with similar levels of success, where those networks have different hidden layer dimensionality. Networks that solved the problem of classification equally well were indeed found (up to the external standard for success imposed in this paper), where the dimensionality of the hidden layer state space was 2, 3, or 4. Perhaps then the physical dimensionality of a state space isn't equivalent to its *semantic* dimensionality, namely, the dimensionality needed to capture essential features of the task domain. In that case, how should we think of or ascertain the relevant semantic dimensions of polytopes?

One option is to assume that there exists some optimal dimensionality at the level of the hidden layer state space for a particular problem, which is set by features of the categorization task combined with peculiarities of the networks studied. This number would correspond to the number of hidden layer units that lead to the 'best' performing networks. In this case, higher dimensional state spaces are redundant whereas those with lower dimensions aren't as good at solving the problem because of a lack of freedom in representing crucial features of the task domain. The number of semantic dimensions is then equivalent to the optimal dimensionality.

Another option is that the semantic dimensions are independent of the physical dimensions of the networks. In this case, one would need to find something similar between the state spaces of networks that are solving the same task, where those state spaces have different numbers of hidden units. This project, which looks for state space similarity at the level of clusters of points in the hidden layer has yet to be explored, and would prove to be a boon for the 'partitioning-of-activation-space' point of view of mental content (Laakso & Cottrell, 2000, p. 54).[2] I will elaborate on this idea in section 6.3.

6.2 Polytopes in the Context of Mutual Information

The existence of regions separated by hyperplanes in the state space of the hidden layer has been noticed before (O'Brien & Opie, 2001, 2006; Churchland, 1995, 2012). What these discussions do not contain is a principled way to think of quantitative regularities that might exist in these regions. I have shown above that different initializations for networks learning a task lead to the same content as determined through the information-theoretic measure of mutual information.

Mutual information represents a natural measure for grounding our understanding of how this works by adverting to the idea of *information transmission*. For some polytope y_m and input class x_n, the central quantity in Definition 1 weights the information y_m carries about x_n by the probability of the occurrence of that pair. One wants a vehicle of content to be informative about its content and this is what the definition employed achieves. Referring to mutual information more than just institutes a statistical measure that relates inputs to

outputs, it attempts to make precise the nature of information flow in the network. It has the additional benefit of doing so through a quantity that measures arbitrary dependencies between input and output, as opposed to, say, the linear dependencies enshrined in Pearson's correlation coefficient (Li, 1990).

In a sense, I have deployed an extension of Usher's proposal regarding mutual information and conceptual representations, in the context of artificial neural networks (Usher, 2001). The output unit's 'role' is to determine which of the input classes that exist in the external world was presented to the network, and in one way of thinking, it does this by consulting the polytope that was activated. The means of determining content through mutual information presented here differs from (one of) Usher's proposals in the insistence of including the pre-factor that multiplies the pointwise mutual information in the definition of mutual information. This ensures that if there are two different inputs that both contribute positively to the pointwise mutual information between an input class and a polytope, the input class with greater joint probability of occurrence will be favoured. In addition, I explicitly specify the constraint that this contribution must be positive, thereby ensuring that the occurrence of the polytope indeed reduces one's surprise about the occurrence of the input class that constitutes its content.

6.3 Future Work

There remains much work to be done from both a modelling and conceptual point of view in order to further advance the long-standing debates tackled in this paper. I'll pause to mention perhaps the most pressing conceptual issue: that of state space similarity for networks with different numbers of hidden units. Laakso and Cottrell (2000) have done some nice work in explicating state space similarity at the level of activation points, one level below the partitions addressed in this paper. There is a clear sense in which their work may apply here. One could compute, for example, pairwise distances between the centres of mass of polytopes in networks with any number of output units say, and then compute Pearson's correlation coefficient between vectors of distances for networks of different hidden layer state space dimensionality. Analogously to the case discussed in Laakso and Cottrell (2000), if that correlation is high, then those networks have partitioned their state spaces in similar ways. I conjecture that a similar result to that obtained by Laakso and Cottrell for activation points, exists for polytopes.

The satisfaction of such a conjecture would go some way in helping us understand how networks with different architectures trained on the same task accomplish similar levels of performance, at the level of explicit partitions in hidden unit activation spaces. The analogue of this accomplishment in the context of the biological systems these artificial neural networks attempt to (crudely) mimic, is the fact that different biological networks (regions of the brain for instance) can share similar informational content while not having identical architecture. Developing an understanding of shared quantitative properties exhibited by polytopes in networks of varied architecture would presumably go some way in helping to explain *how* these networks function, and would form an integral part of any general theory that attempts to explain cognitive behaviour and development by referencing partitions in hidden layer state spaces.

Notes

[1] There may exist more nuanced situations where one might wish to invoke selective disjunctions over input classes to determine which of these disjunctions has the highest contribution to the mutual information for any given content-bearing vehicle. In this case, at

least in principle, it may be that one can find disjunctions that are more informative when compared with a single input class. Investigating this suggestion in the context of neural networks (and perhaps more broadly as well) remains work for future investigation.

[2] I thank Gerard O'Brien for stressing the importance of this issue.

Acknowledgements

I thank Dominic Murphy for helpful discussions, as well as Gerard O'Brien for his thoughts on the general context of the problem developed here. Datasets for the sonar classification task and the cardiac SPECT imaging classification task were obtained from the UCI Machine Learning Repository. Work at Sydney was funded by a Philosophy of Neuroscience Postgraduate Scholarship.

References

Bache, K., & Lichman, M. (2013). *UCI Machine Learning Repository*. University of California, Irvine, School of Information and Computer Science, Irvine, CA, USA. http://archive.ics.uci.edu/ml/

Churchland, P. M. (1989). *A neurocomputational perspective: The nature of mind and the structure of science*. Cambridge, MA: MIT Press.

Churchland, P. M. (1995). *The engine of reason, the seat of the soul: A philosophical journey into the brain*. Cambridge, MA: MIT Press.

Churchland, P. M. (1998). Conceptual similarity across sensory and neural diversity: The Fodor/Lepore challenge answered. *The Journal of Philosophy, Inc.*, 95, 5–32.

Churchland, P. M. (2012). *Plato's camera: How the physical brain captures a landscape of abstract universals*. Cambridge, MA: MIT Press.

Churchland, P. S., & Sejnowski, T. J. (1990). Neural representation and neural computation. *Philosophical Perspectives*, 4, 343–382.

Cottrell, G. W. (1991). Extracting features from faces using compression networks: Face, identity, emotion, and gender recognition using holons. In D. S. Touretzky, J. L. Elman, T. J. Sejnowski & G. E. Hinton (Eds.), *Connectionist models: Proceedings of the 1990 summer school* (pp. 328–337). San Mateo: Morgan Kaufmann Publishers, Inc.

Cover, T. M., & Thomas, J. A. (1991). *Elements of information theory*. New York: John Wiley & Sons, Inc.

Dretske, F. I. (1981). *Knowledge and the flow of information*. Cambridge, MA: MIT Press.

Eliasmith, C. (2000). How neurons mean: A neurocomputational theory of representational content (Doctoral dissertation, Washington University in St. Louis, 2000).

Fodor, J. A. (1984). Semantics, Wisconsin style. *Synthese*, 59, 231–250.

Gorman, R. P., & Sejnowski, T. J. (1988). Analysis of hidden units in a layered network trained to classify sonar targets. *Neural Networks*, 1, 75–89.

Kurgan, L. A., Cios, K. J., Tadeusiewicz, R., Ogiela, M., & Goodenday, L. S. (2001). Knowledge discovery approach to automated cardiac SPECT diagnosis. *Artificial Intelligence in Medicine*, 23, 149–169.

Laakso, A., & Cottrell, G. (2000). Content and cluster analysis: Assessing representational similarity in neural systems. *Philosophical Psychology*, 13, 47–76.

Li, W. (1990). Mutual information functions versus correlation functions. *Journal of Statistical Physics*, 60, 823–837.

Lombardi, O. (2005). Dretske, Shannon's theory and the interpretation of information. *Synthese*, 144, 23–39.

O'Brien, G., & Opie, J. (2001). Connectionist vehicles, structural resemblance, and the phenomenal mind. *Communication and Cognition*, 34, 13–38.

O'Brien, G., & Opie, J. (2006). How do connectionist networks compute? *Cognitive Processing*, 7, 30–41.

Ramsey, W., Stich, S., & Garon, J. (1990). Connectionism, eliminativism, and the future of folk psychology. *Philosophical Perspectives*, 4, 499–533.

Rieke, F., Warland, D., de Ruyter van Steveninck, R., & Bialek, W. (1997). *Spikes: Exploring the neural code*. Cambridge, MA: MIT Press.

Rumelhart, D. E., McClelland, J. L., & the PDP Research Group. (1986). *Parallel distributed processing: Explorations in the microstructure of cognition, Volume 1: Foundations*. Cambridge, MA: MIT Press.

Sejnowski, T. J., & Rosenberg, C. R. (1987). Parallel networks that learn to pronounce English text. *Complex Systems*, 1, 145–168.

Shannon, C. E. (1948). A mathematical theory of communication. *The Bell System Technical Journal*, 27, 379–423, 623–656.

Shannon, C. E., & Weaver, W. (1949). *The mathematical theory of communication*. Urbana: University of Illinois Press.

Shea, N. (2007). Content and its vehicles in connectionist systems. *Mind & Language*, 22, 246–269.

Shea, N. (2013). Naturalising representational content. *Philosophy Compass*, 8, 496–509.

Usher, M. (2001). A statistical referential theory of content: Using information theory to account for misrepresentation. *Mind & Language*, 16, 311–334.