The extreme structural regularity across the expanse of neocortex suggests there is a canonical cortical algorithm/circuit, defined at macrocolumnar scale (e.g., 300-500 um across), operative across all modalities and all hierarchical levels, which underlies general intelligence, performing the core functions of memory and retrieval.[1] This includes both memory/retrieval of unique experiences, i.e., *episodic memory*, and of similarity/class (statistical) structure, i.e., *semantic memory*, as well. My primary research goal has been to understand this canonical algorithm.[2] My model of this cortical algorithm/circuit is called Sparsey because its most essential property is that items of information (concepts) are represented by *sparse distributed representations* (SDRs), i.e., relatively small sets of co-active binary units (principal cells) chosen from much larger populations. [N.b.: SDR ≠ "sparse coding".[i]] An SDR can be seen as a type of Hebbian *cell assembly*. I consider the L2/3 portion of a macrocolumn (e.g., V1 hypercolumn, barrel-associated column) to be the most likely candidate for such a population (the L5 portion might also constitute such population). I refer to these populations as "coding fields" or as "macs" (since, I consider the L2/3 portion to be the primary information repository of the macrocolumn, with the other layers in subordinate roles). I've further speculated (Rinkus 2010) that the L2/3 pyramidals comprising a minicolumn function as WTA competitive modules (CMs), as in Fig. 1.[3]

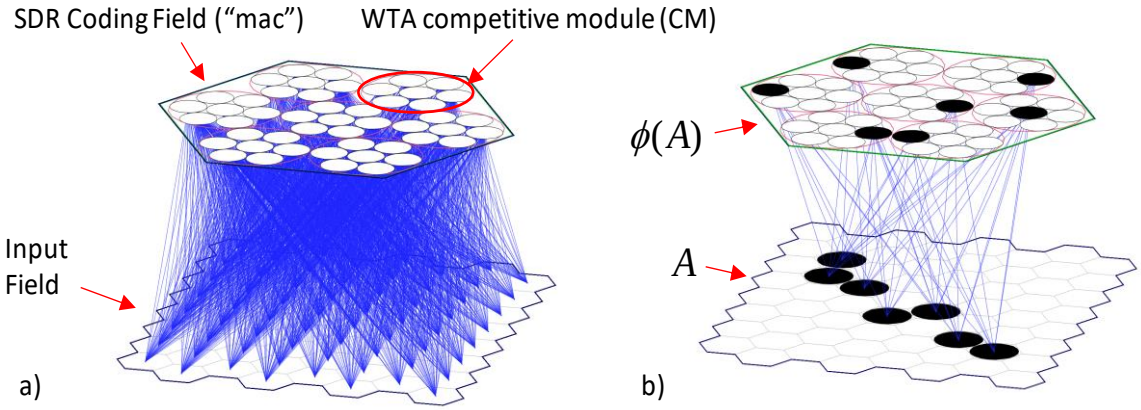


Figure 1: a) An example SDR coding field ("mac") consisting of Q=7 WTA competitive modules (CMs), each with K=7 binary units. An input field of binary neurons is completely connected to the coding field. b) A particular input A, its code, ϕ(A), consisting of Q active (black) neurons, and the *bundle* of weights that would be increased to form the association are shown.

Using ballpark assumptions (e.g., Peters & Jones) of, say Q=70, minicolumns per macrocolumn, and say, K=20 L2/3 pyramidals per minicolumn, then any given SDR will consist of 70 active pyramidals, one per minicolumn, and the code space of the coding field will be 20^{70}. While this code space is vast, the actual number of SDRs ("codes") stored in such a coding field is limited by other factors and probably only a tiny fraction of the code space, e.g., hundreds to thousands. Each of these codes will represent a particular (possibly spatiotemporal) input to the

[1] This page discusses why we don't yet have clear evidence for macrocolumn-scale units in more frontal regions.

[2] I also want to generalize this model to hippocampus and analogous structures in other species, e.g., mushroom bodies, one fundamental organizing rubric being how these memory-related structures/systems relate to the spectrum from code separation to code completion. Eventually, also to basal ganglia, amygdala, cerebellum.

[3] The WTA functionality is modeled at an abstract level; the explicit inhibitory infrastructure that would subserve it is not explicitly modeled, though also a desired target of future research.

coding field, e.g., pattern A in Fig. 1b. These codes are the long-term (permanent) memory traces of experienced inputs. As discussed in this recent Medium article, they can be formed on the basis of a single trial. In my overall theory, it is not the *strengths* of the synapses afferent to an SDR (cell assembly) (blue lines in Fig. 1b), but their resistances to passive decay ("*permanences*"), that rapidly increase with subsequent hippocampally-assisted replays, i.e., during *consolidation*, making the memory permanent.

The foregoing already reveals or suggests many ways in which my theory of cortical computation differs from mainstream machine learning (ML), including deep learning (DL), and also from the majority of prior computational neuroscience models. It's worth stepping back and listing some of these key differences. Elaborating on these differences and their consequences is kept to a minimum within the list, but references/links are given.

1. For Sparsey, memory functionality (storage and similarity-based retrieval) is primary and optimization of some global objective function, e.g., loss on classification metric, is secondary. In contrast, for the vast majority of ML/DL models, optimization is primary and memory, per se, has been an afterthought. That is, by design, such models learn only enough information to support targeted classifications or to support probabilistic inference (e.g., similarity-based search/prediction). Memory of specific instances (episodic memory) has been essentially ignored: only in the last 5-6 years has episodic memory functionality been added to DL models (e.g., Neural Turing Machine, Neural Episodic Control, Memory Networks). But even within computational neuroscience proper, learning is often characterized as optimization (e.g., sparse coding models). This difference has many implications, some of which are discussed in more detail in later sections as well as in this abstract and the Medium article.
2. As noted above, an SDR (cell assembly) is assigned in a single event. This, in turn, provides support for the idea that increases to excitatory synapses from one principal cell to another may be essentially binary. This contrasts sharply with optimization-centric ML/DL models, which have generally viewed synaptic weights as changing [both up and down) in tiny deltas (and based on numerous trials, in an offline (batch) learning setting]. But, I think the prevailing view among neuroscientists, owing to a long legacy of LTP studies, often using artificial stimulation protocols, and generally focusing on single synapses, not whole bundles of synapses between two cell assemblies, might also lean towards synaptic change occurring in small deltas over many trials. I think we are only now entering a period where experimental methods, e.g., calcium imaging, VSDs, might be able to directly address such questions, and I would like to be involved with such experiments.
3. As noted, Sparsey proposes that the L2/3 pyramids of a single minicolumn function as a WTA competitive module (CM). Of course, this is a very strong assumption. It is also eminently falsifiable, though to my knowledge, no experiment has directly tested this hypothesis. Again, methods are only now approaching the ability to do it, though there are still some hurdles. Also note that while Sparsey hypothesizes that the normal mode for the competition with a CM is 1-WTA, it also hypothesizes a k-WTA (e.g., k=2) mode as well, which is further fodder for experimentation [and in particular, would relate to recent ideas about working memory (WM), e.g., transient and sustained cells (Warden and Miller 2007)] (see Section 7).
4. As its fundamental unit of meaning is an SDR (cell assembly), Sparsey strongly departs from the *functional* "Neuron Doctrine", in which the central unit of meaning is the single

neuron. In fact, it implies a novel theory for how the classical, typically unimodal (Gaussian/Gabor) tuning functions of single cells emerge as a side-effect of the involvements of single cells in a series of SDRs. This idea is elaborated in Section 6.

5. A fundamental property of Sparsey is that it has an efficient way to map more similar inputs to more similar (more highly intersecting) SDRs. This is achieved by: a) a computationally simple method for a mac to estimate the "familiarity" of its input (which is normalized to the interval [0,1]); and b) an (also computationally simple) way of adding into the SDR selection process, an amount of randomness (*noise*) that is inversely proportional to the familiarity. To my knowledge, this concept for the (normative) usage of noise remains novel within the entire neuroscience literature. A sketch of neuromodulator-based implementation of this familiarity-contingent noise was given in (Rinkus 2010), and is discussed further in Section 4. I would like to be involved with a course of experimentation addressing this question.
6. Sparsey's algorithm for choosing SDRs (in a way that preserves similarity), the *Code Selection Algorithm* (CSA), includes many novel, strong, testable neural hypotheses, including:
   a) This fundamental algorithm, the CSA, executes once in a local (e.g., to given mac, more specifically, to that mac's L2/3 portion) gamma, cf. (Fries, Nikolic et al. 2007, Fries 2015). But the CSA implies that there are *two consecutive rounds of competition* amongst the competing principal cells, the first to compute the aforementioned "familiarity" signal, and the second to choose the final winner (in each of the Q CMs) and thus active SDR on that gamma cycle. Since a gamma cycle is ~25-40ms long, this is a highly provocative claim (motivated from the normative direction), though I contend well within neuronal timing constraints, which again, is eminently falsifiable.
   b) Following on the last point, the CSA implies a fast-time-scale (~10ms) modulation of the principal cells' nonlinearity based on a global familiarity (inverse novelty) of the macrocolumn's overall input. My working hypothesis is that this is mediated by some combination of fast (i.e., 100ms timescale) transient ACH and NE signals. To my knowledge, such a fast modulation of the neuronal nonlinearity to achieve a particular functional goal—to shift the coding field's (macrocolumn's) dynamics along a spectrum from code separation to code completion [cf. (Curto, Itskov et al. 2013)]—has not been previously described. I would like to explore whether this principle operates in hippocampus as well.
   c) A learning scheme in which, whenever a pre-post coincidence occurs, the synapse's weight is set to its maximal value (e.g., binary "1") *and* another property, the synapse's *permanence* (i.e., resistance to passive decay), is modulated on the basis of the recent history of the synapse. This is discussed further in Section 1 below. But, in broadest terms, it reflects a paradigm-shift:
      i. away from the slow, small-delta learning concept germane to optimization-centric, gradient-based learning (which continues to dominate ML/DL) in which the effect of every individual synaptic weight upon a global objective is explicitly, and repeatedly (at least in every epoch) computed,
      ii. towards a fast (i.e., single-trial) large-delta unsupervised learning concept where the "atomic" learning event is a *set* of correlated changes to a large

number (bundle) of synapses from one ("pre-synaptic") SDR to another ("post-synaptic") SDR. Also, the pre and post-synaptic SDRs can be in the same coding field, i.e., when coding fields are recurrently connected, which mediates formations of chains of SDRs, i.e., Hebbian phase sequences. Also note that although I use the phrase "large-delta", the absolute magnitude of the change in synaptic strength can be small [due to the fact that it is the sum of many approximately simultaneously arriving (single, i.e. first) spikes along numerous afferents that yields a strong signal] See this abstract submitted to COSYNE 2019 for more detail, and n.b., I believe I've described a novel concept for how (waves of first) spikes are used to communicate between cell assemblies [in particular, as the abstract notes, this concept is strictly stronger than the scheme described in Sec. 7.2.2 of Gerstner et al's book.]

There are many more opportunities for research in addition to those given in the preceding list. Some are discussed in the following sections.

1. HOW DOES THE BRAIN CREATE PERMANENT MEMORY TRACES OF SPECIFIC EVENTS, I.E., EPISODIC MEMORY TRACES?

By definition, a "specific event" happens once, so these traces are formed on the basis of single trials. The prevailing view is that a strong trace is immediately formed in hippocampus and that replays of that trace, e.g., during sleep, gradually strengthen, or consolidate, a neocortical trace over time. The immediate hippocampal trace can fade as the neocortical trace solidifies. In this prevailing view, the neocortical synapse is typically characterized by a single variable, it's weight, which increases gradually across replays. Metaplasticity is modeled not by an explicit, additional parameter of the synapse, but by a global parameter, learning rate, which decreases throughout training.

In contrast, in Sparsey, synapses have an additional parameter—specifically, resistance to passive decay, which I call *permanence*—that is also modulated. Upon the first occurrence of a particular input, the weights involved in both the hippocampal and neocortical traces are increased to their maximal values. But the *initial* neocortical synapse passive decay rate is high compared to the fixed decay rate of hippocampal synaptic weights. Thus, in a subsequent consolidation period, the relatively more stable hippocampal trace can replay multiple times, guiding the replay of the neocortical trace, during which the decay rates of the neocortical synapses decrease (permanences increase). The cortical trace rapidly becomes permanent across those replays. But, the fixed hippocampal decay rate allows the hippocampal trace to gradually fade away or be overwritten by new experiences.

Sparsey's Permanence is similar in concept to *metaplasticity* in Fusi et al's Cascade model, but its dynamics, described in Rinkus (2014), is simpler. It relies on the idea that for inputs generated by natural domains, i.e., domains populated by entities with recursive compositional (part-whole) structure, pre-post activation coincidences arising from structural regularities of the domain will occur exponentially more frequently than coincidences due to noise or spurious alignments. This suggests that we will be able to find parameter settings, e.g., the fixed hippocampal permanence, the schedule of increase of neocortical synapse permanence, etc., that will tend to preferentially embed permanent neocortical traces of the domain's structures (both

spatial and temporal), while allowing traces of spurious inputs to fade away. I also emphasize that Sparsey's permanence method is purely local [involving only keeping track of time since last weight increase (i.e., since last pre-post coincidence)] and computationally far simpler than two other recent solutions to *catastrophic forgetting* (or, to Grossberg's "Stability-plasticity" dilemma), the Elastic Weight Constraints (EWC) model (Kirkpatrick, Pascanu et al. 2017), and the Memory Aware Synapses (MAS) model (Aljundi, Babiloni et al. 2017), both of which require repeated/continual evaluation of each weight's importance to (the constantly increasing number of) previously learned tasks/mappings.

2. EXTREMELY EFFICIENT LEARNING AND BEST-MATCH RETRIEVAL:

Sparsey's core algorithm, the *Code Selection Algorithm* (CSA), is an extremely efficient algorithm for the fundamental cognitive processes of learning (storage of) new information and best-match retrieval of stored information. It turns out that SDR admits learning and best-match retrieval (more generally, probabilistic inference) methods that do not involve computing gradients or MCMC and are thus far more efficient than those of mainstream, optimization-centric ML/DL/GPM approaches. In fact, Sparsey does both learning and best-match retrieval in *fixed* time, i.e., the number of operations needed remains constant as the number of items stored in the memory increases (shown in my 1996 thesis and since, e.g., 2010, 2014, and 2017). Relatedly, a recent Science paper, "A neural algorithm for a fundamental computing problem" (Dasgupta, Stevens et al. 2017). describes the similarity-based search that occurs in the fly olfactory system as a novel and important version of locality-sensitive hashing (LSH), which is a class of algorithms that can retrieve the best-matching stored item in fixed time. However, Sparsey learns (with single trials) the locality-sensitive hash from the data, i.e., learns the similarity (statistical) structure of the input space, whereas the Dasgupta et al model does not. (I've confirmed this with the author; he was not aware of my work but agreed to cite it in future papers.) Nevertheless, their mapping of their model to neural structures and fundamental cognitive processes is important and something I would like to continue pursuing for Sparsey as well, if possible, in collaboration with Dasgupta and his colleagues. Insofar as Sparsey can be viewed as doing adaptive LSH, it also qualifies as a method to do on-line learning of indexes into databases. As remarked in the recent paper, "The Case for Learned Index Structures" (Kraska, Beutel et al. 2017), the ability to do learning, and moreover, on-line learning, of indexes (including multidimensional indexes) into massive conventional databases has massive economic implications. Finally, while Sparsey achieves the goals of LSH, it is in fact far simpler and more efficient than standard LSH (Indyk, Motwani, Andoni, Razenshteyn).

3. A NOVEL EXPLANATION OF THE REPRESENTATION AND USE OF PROBABILITIES IN THE BRAIN:

It is instructive to compare Sparsey to Hinton's recent "Capsules" model, wherein each *capsule* represents a particular object. This is localist: a field of N capsules can represent N objects. The instantaneous activity vector over the capsule's units represents the settings of the features (latent variables) of the object and the overall length of the vector represents the probability of the presence of the object. Sparsey has a completely different and I believe far more powerful, general, and elegant treatment of probability. All codes stored in a Sparsey coding field are of size Q, where Q is the number of minicolumns comprising the coding field. And, the learning algorithm causes more similar inputs to be mapped to more highly intersecting codes. Thus, if one grants

that similarity (in the input space) is reasonably correlated with likelihood, then the likelihood of any input X stored in a coding field is measured by the fraction of X's code that is currently active (as part of the full set of *Q* units that is currently active). This concept is illustrated in Figure 2 (portion above dashed line). Reading down the first column, the inputs B-E have progressively decreasing pixel overlap with A (shown in second column). Correspondingly, the SDR codes (4th column), $\phi(B)$ to $\phi(E)$, have progressively smaller intersection with $\phi(A)$.

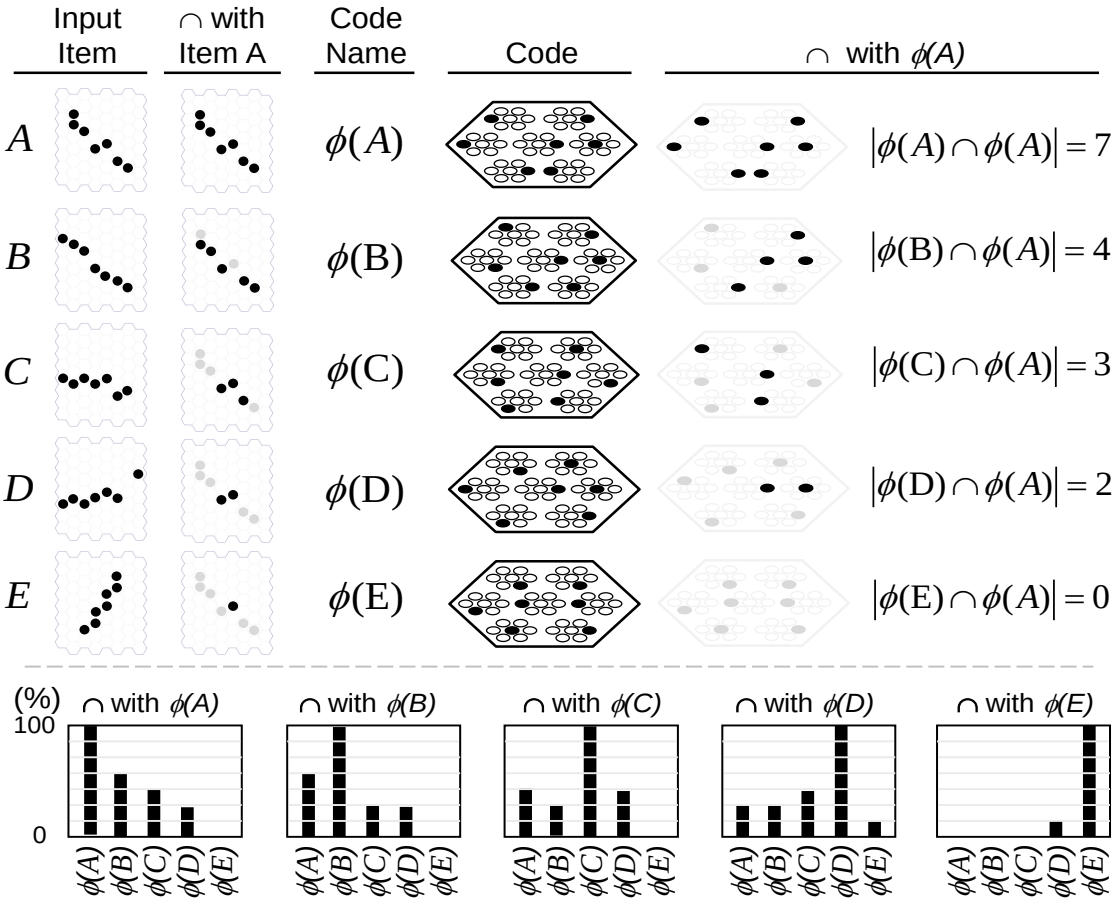


Figure 2: (Above dashed line) Illustration of principle of mapping similar inputs to similar SDR codes. See text. (Below dashed line) Illustration of how the probability/likelihood of an input can be represented by the fraction of its code that is active. When ϕ(A) is fully active, the hypothesis that input A is present can be considered maximally probable. Because the similarities of the other inputs to the most likely input, A, correlate with their codes' overlaps with ϕ(A), the likelihoods of those inputs are represented by the fraction of their codes that are active. In the intersection ("∩") columns, black indicates units intersecting with either the input pattern A or its code, ϕ(A); gray indicates non-intersecting units.

The beauty of this principle is that the likelihoods of ***all*** inputs stored in the coding field are *simultaneously* physically represented by the fractions of their codes that are physically active. The macrocolumnar code activated in response to an input simultaneously represents both:

a) the likelihood of the *single* exactly (or best) matching previously stored input, ***and***
b) the *entire likelihood distribution* over *all* inputs (hypotheses) stored in the coding field.

The charts below the dashed line in Figure 2 show this for each of the five inputs' codes being fully active. And, since these fractions of codes (which are sets of coactive units) are all physically active in superposition, they can all materially influence downstream computations (with strength proportional to the fraction active). In contrast, a capsule's activity vector's length has a single value at any one moment and thus can represent only one probability (likelihood) value (not an entire distribution) at any one time. This aspect of my theory is elaborated in a 2017 arxiv paper, "A Radically new explanation of how the brain represents and computes with probabilities", where I contrast it with the long-standing probabilistic population coding (PPC) theories, e.g., Georgopolous, Pouget, Movshon, others. A discussion of the relation of Sparsey to Hinton's Capsules is here.

Finally, since an invocation of Sparsey's core algorithm, the CSA, chooses (activates) a code and since that code qualifies as a distribution over all hypotheses stored in a mac, the CSA can be viewed as doing *belief update*. *Crucially, since the CSA executes in time that remains constant as the number of stored hypotheses increases, the CSA does "belief update" in constant time, a capability which has not been reported for any other model and which has massive implications regarding scaling.*

4. A NOVEL USAGE OF NOISE IN COMPUTATION:

As noted earlier, a Sparsey SDR coding field is a set of Q WTA competitive modules (CMs), each consisting of K binary units: thus, an SDR code is a set of Q units, one per CM. Amongst many novel aspects, Sparsey's method of choosing / activating a code in response to an input—the aforementioned CSA—involves a novel concept for the use of noise for the purpose of ensuring similarity preservation. A code is chosen as Q independent draws, one in each CM, where the distribution from which the draw is made reflects the degrees of support of the K competing units, which in turn reflects prior learning. Clearly, if the input has been seen before we want to reactivate the same code that was activated on the original presentation; i.e., we want to ensure that the same winner is chosen in each of the Q CMs. On the other hand, to the extent that the input is novel (unfamiliar), we want to activate a new code having decreasing intersection (increasing Hamming distance), on average, with all previously stored codes; i.e., we want to choose more randomly in each CM, thus pushing the average Hamming distance with previously stored codes towards chance (i.e., code separation). The novel use of noise in Sparsey is simply to add noise to (flatten) the distributions (in each CM) in proportion to the novelty of the input. Technically, Sparsey computes a ***familiarity***, i.e., *inverse novelty*, signal, $\boldsymbol{G}$, and minimizes noise in proportion to G. This achieves the desired dynamics of causing the expected intersection of the SDR code, $\phi(X)$, activated in response to a current input X with any previously stored code $\phi(Y)$ to vary directly with similarity(X,Y): this page has a useful dynamic explanation of this principle. My working hypothesis (described in 2010) is that NE or ACh, or some combination of these (and perhaps other neuromodulators) implements this familiarity-contingent noise modulation and I've long wanted to work with people with expertise in neuromodulators to explore this hypothesis.

5. HIERARCHICAL COMPOSITIONALITY, GRADUATED PERSISTENCE, CHUNKING:

As noted above, Sparsey does single-trial, on-line learning of spatiotemporal (sequential) input patterns, embedding chains of SDRs (essentially Hebbian phase sequences of "cell assemblies") as *episodic* memory traces of the inputs. However, it is also essential that Sparsey is hierarchical, allowing arbitrarily many levels, where each level consists of multiple coding fields (proposed as analogs of cortical macrocolumns, or "macs") and the time constant (persistence) of the codes (SDRs) increases (typically, doubles) with level. Thus, more precisely, Sparsey embeds *hierarchical*, spatiotemporal memory traces using both *chaining* within levels and *chunking* between levels, e.g., a single code in a level J mac can become associatively linked with two successive codes in a level J-1 mac, i.e., a form of compression (see page, page, and page for more details). While informational items are represented sub-symbolically as SDRs, I am deeply interested in explaining how progressively higher-level informational structures/symbols, e.g., phonemes, syllables, morphemes, words, etc., and more generally, recursive part-whole compositions, automatically emerge from the more or less continuous, multimodal input stream. The fact that the generalities that automatically emerge in Sparsey are represented by patterns of intersections over SDRs (more generally, over chains of SDRs, and more generally still, over hierarchical chains of SDRs) of *particular* experienced inputs, suggests that Sparsey may have greater power for modeling/explaining the often quite *idiosyncratic* nature of classes, rules, or schemas learned and used by people, e.g., in language learning, cf. overgeneralization in children.

The recursive compositional structure of natural objects and events vastly reduces the number of samples needed to learn good models of natural domains. Thus, if we factor the recognition problem into a sequence of scale-specific sub-problems (e.g., carried out at the different levels of a hierarchical network), the number of samples needed to train each scale might be small and the number of samples needed overall might be exponentially smaller than for the unfactored "flat" approach, cf. Poggio and colleagues' 2012 description of the hierarchical ventral stream as accomplishing this reduction of *sample complexity*. Empirically establishing this claim is another target for future experimental work. See this abstract accepted to NIPS 2018 Continual Learning Wkshop for further details.

6. THE CLASSICAL LOW-COMPLEXITY TUNING FUNCTION OF A SINGLE NEURON IS AN *ARTIFACT* OF ITS PARTICIPATION IN MANY SDR CODES OF SPECIFIC INPUTS:

I maintain that the classical, Gaussian/Gabor-like single unit tuning function (TF) is an artifact of experimental limitations, i.e., of approximately 60 years of single/few-unit electrophysiology, mostly using massive averaging and low-complexity probe stimuli. Of course, there is a growing body of results using more detailed probes, less averaging, etc., supporting far more heterogeneous ("mixed-selectivity") TFs, e.g., (Rust and DiCarlo 2010, Nandy, Sharpee et al. 2013, Fusi, Miller et al. 2016, Yamins and DiCarlo 2016). One of my research goals is to explain the classical unimodal single unit TF as simply reflecting the history of a unit's participation in SDR codes. The result can be seen almost immediately from first principles. If you map (in particular, with single trials) inputs to SDRs in a way that preserves similarity, then as noted earlier, the similarity structure (higher order statistics) of the inputs will automatically emerge in the pattern of intersections over the SDR codes. If you then probe a single unit using low complexity stimuli (e.g., oriented edges whose spatial constants are large with respect to the unit's receptive field) then a reverse correlation process will show a classical-looking TF, i.e., with one/few modes on one/few directions in the unit's high-dimensional input space.

Figure 4 shows preliminary supporting evidence from a simulation involving the model of Figure 4a presented with sequences, a small subset of which is shown in Figure 4b. Figure 4c shows a sample of TFs for 40 cells selected from various of the 16 L1 (analogous to V1) macs. The number of codes in which a cell participated is given under its TF. The primary result visible in Figure 4c is that these single-unit TFs have a great deal of higher-order structure and in some cases have some resemblance to classical V1 cell TFs. We underscore this resemblance by overlaying several TFs from one of Hubel & Wiesel's seminal papers (Hubel and Wiesel 1962), directly on top of our model's single-cell surrogate TFs. Furthermore, some of them evince multi-selectivity (Fusi, Miller et al. 2016). For example, the top left cell looks like it has been included in codes activated in response to vertical edges along the left side of the field and to diagonal (135°) edges. Similarly, the fifth cell in the second row looks like it has been included in codes for vertical edges occurring in the left and right halves of the field.

The TFs in Figure 4c are semi-quantitative as we have not computed their statistical significance, and so must be considered preliminary. However, the deviation from the more homogenously gray fields that would be produced by chance, especially for higher cell usage counts, e.g., 20-50, strongly suggests that cell win probability (within its CM) is being influenced by pairwise and higher-order pixel co-occurrences, or in other words, that cells are acquiring sensitivity to *features*, or to multiple features, in the input space. Again, we emphasize that all cells' TFs are perfectly flat initially and that the surrogate TFs in Figure 4c emerge during learning via incremental superposition of *en masse* Hebbian updates to whole SDR codes based on single trials. This suggests a novel explanation for the formation of the classical single-cell TFs, that have been observed in cortex for 60+ years.

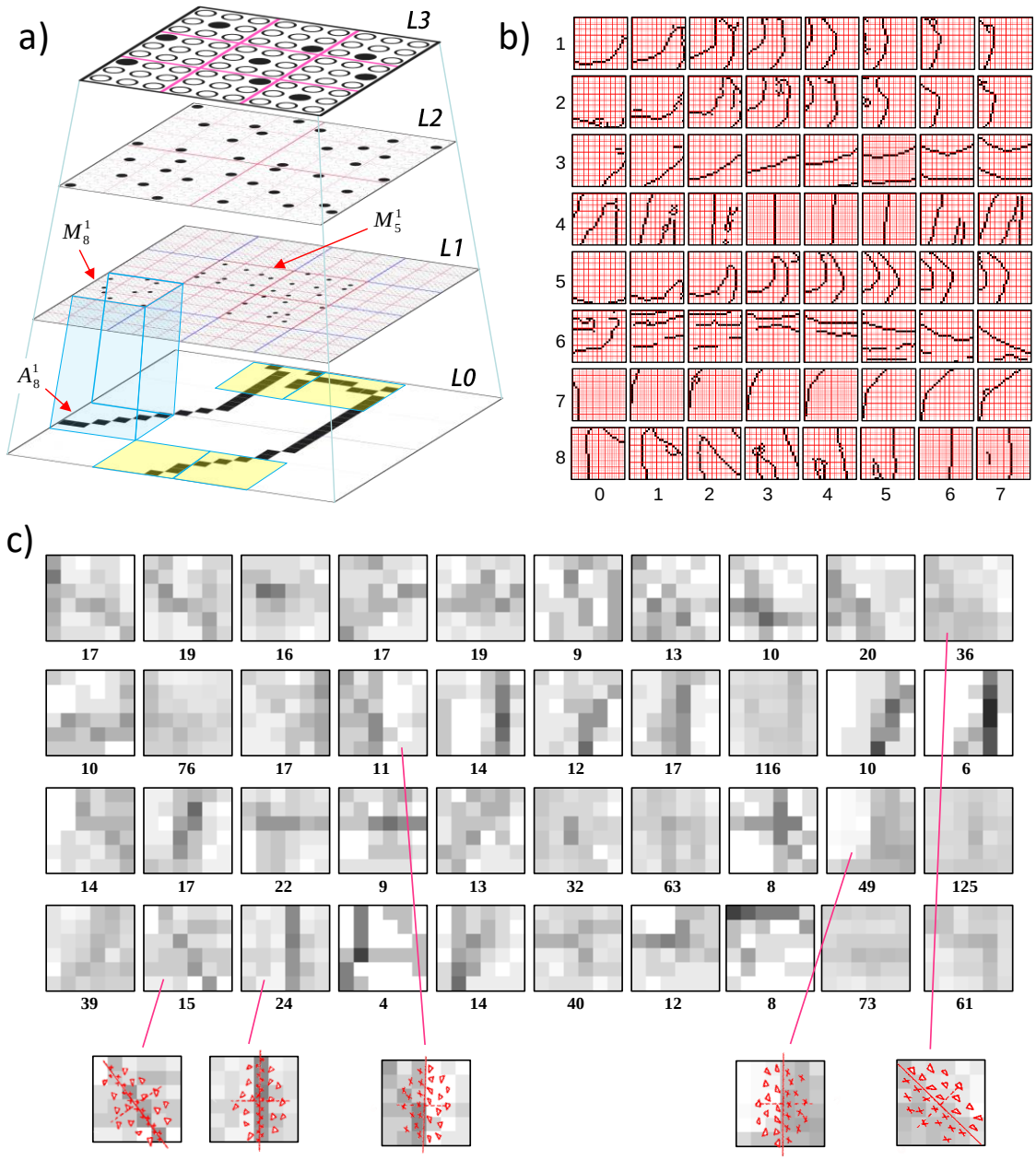


Figure 4: a) 4-level Sparsey model with a 24x24 binary input level (L0), 4x4 sheet of "V1" macs (L1), each with a 6x6-pixel receptive field. Two of the L1 macs are indicated, M_8^1 and M_5^1, as is M_8^1's RF (or "aperture"), A_8^1. b) Sample of the sequences presented to the model during training. (Note: the bolder red hatching in some frames is a visual artifact.). c) Reverse correlations showing depictions of the tuning functions (TFs) of several of the model's V1 cells (from various of the 16 V1 macs) showing a resemblance to classically observed single-cell TFs. At bottom, we overlay TFs from Hubel & Wiesel's 1962 paper (red) on some of our model cell TFs to emphasize the resemblance. The number of SDR codes in which a cell participated [and thus, over which the "reverse correlation" (averaging, normalizing and representing in grayscale) is performed] is shown below each TF.

These ideas tie in directly with the concepts of multitaskers, mixed selectivity and diversity, as developed by Fusi, Miller, and others. Figure 5 further elaborates on Sparsey's explanation of these concepts. At the top, two units, a and b, happen to be included, e.g., by chance, in the SDR, $\phi(E_1)$, of the first input given to this small model. At that point, not only a and b, but all six units comprising $\phi(E_1)$, have exactly the same TF; they are completely correlated. As future inputs occur, a and b may sometimes be included together in a new code, but in general (reading down the figure), they will have progressively diverging histories of code inclusions, and thus, diverging TFs (they become decreasingly correlated). Thus, units (even within the same CM) will develop different (*diverse*) tuning functions, i.e., mixed selectivities. NOTE: This figure does not actually display the property of preserving similarity from input space to code space. Thus, the final TFs at bottom left and right do not have the classical, unimodal appearance. However, the TFs of Figure 4c, resulting from a simulation in which similarity was preserved, do show the uni/few-modal TF structure. One of my near-term goals is to further establish this SDR-based, i.e., cell-assembly-based explanation of classical single cell TFs, both spatial and spatiotemporal. In particular, the simulations of Figure 4c were too small for metaplasticity (described in earlier sections) to have an effect: but presumably, by acting to reduce the presence of outliers (due to either noise or spurious alignments), it would make the resulting TFs (shown via reverse correlation) even more classical looking.

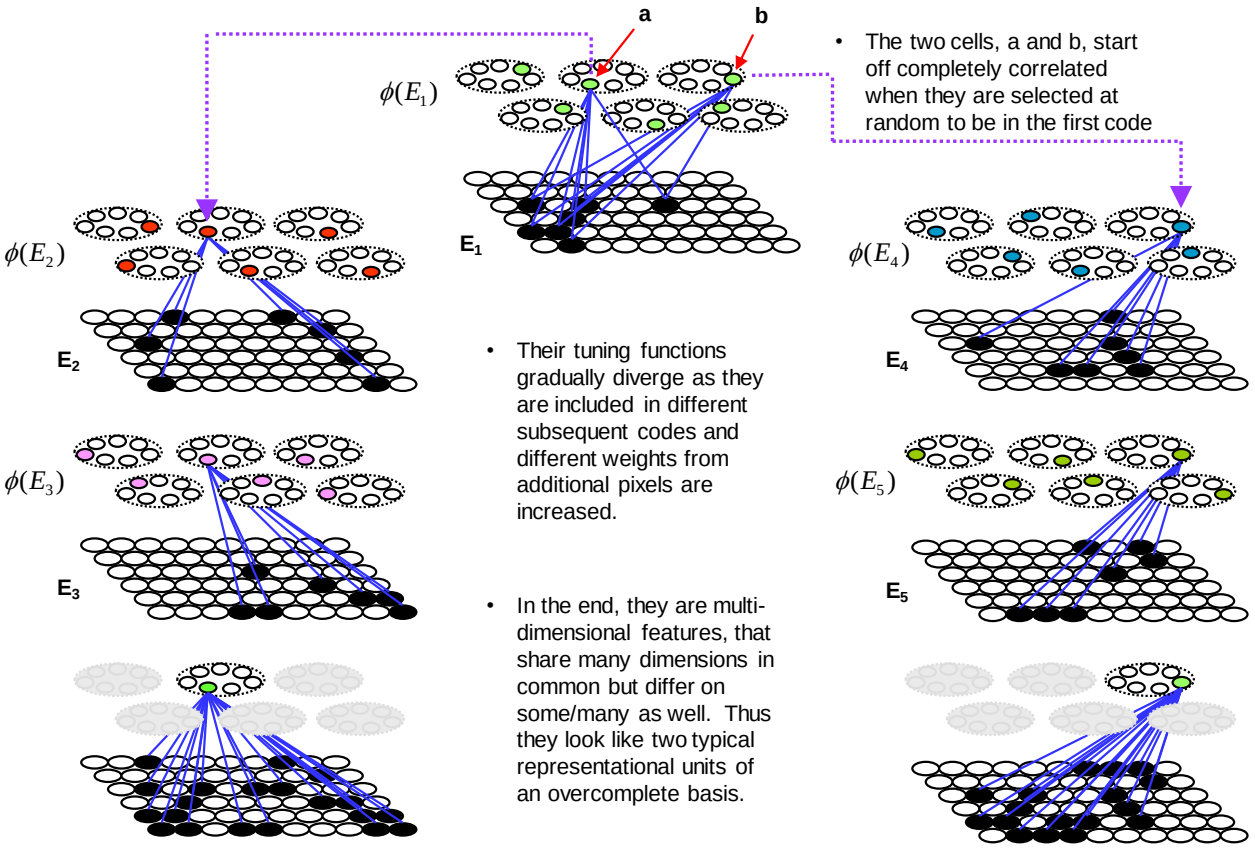


Figure 5: Illustration of diverging tuning functions (TFs) depending on different histories of inclusion in future SDR codes. Here the mac has Q=6 CMs, each with K=7 units. Two units, a and b, by chance, are included in the SDR, $\phi(E_1)$, of some initial input, E_1. The bottom-up (U)

weights from *all* active pixels to *all* Q=6 active units (light green) are increased. At this point, all six units comprising $\phi(E_1)$ are 100% correlated: they have the exact same TFs. But, then a second pattern, E_2, presents. Unit a happens to be included in E_2's code, $\phi(E_2)$, but unit b is not. Thus, some new U weights to unit a increased, but not to unit b. As various future inputs present, unit a and unit b will have different histories of inclusion in the codes of the inputs: thus, the TFs of a and b gradually diverge over time. Though not shown in this figure, if the learning algorithm that picks SDRs for inputs preserves similarity, e.g., in this case, pixel-wise overlap, then the set of inputs in whose codes a given unit is included will have some increased similarity compared to across sets.

7. NOVEL EXPLANATION OF WORKING MEMORY / STM:

Across disciplines, the prevailing view of working memory (WM) has been that the different items stored in WM are stored in physically disjoint fields, e.g., O'Reilly and colleagues' PBWM model in theoretical neuroscience, and Waibel et al's TDNN in machine learning.

A paradigm shift in understanding WM becomes possible when we think of items as represented by SDRs. In this case, the state of WM (over the several recent items) manifests as certain patterns of change over the SDR code active in a *single* coding field. This concept and mechanism, called overcoding-and-paring (OP), has in fact been patented (US Pat: 8,983,884). The idea is basically that when the first item of a sequence presents, an "overcode" is activated in response. The overcode is still an SDR but with (nominally) twice as many cells active. Specifically, we implement this by making the CMs of a mac function in 2-WTA mode for the first sequence item (resulting in 2Q active units, two per CM). When the second item comes in, the CMs change to a 1-WTA mode. That second input item causally influences how that overcode is "pared down" to a regular-sized SDR. But crucially, the subset of cells that remains active after the second item enters is a *subset* of the set of cells originally activated on the first item. This is crucial because it means that the pared-down code formally depends on both sequence items despite having activated (as a subset of an overcode) for the first item. There is no violation of causality and also no requirement to explicitly buffer items temporarily in disjoint coding fields. There is some evidence consistent with this new view of WM, e.g., "The representation of multiple objects in prefrontal neuronal delay activity" (Warden & Miller, 2007).

As an analogy, OP is akin to carving a statue from a block [all the matter is there at the outset, a subset remains at the end, but that subset depends on the extent of the initial block (i.e., the final statue can't include parts not contained in the initial block)]. In contrast, prior buffer-based mechanisms for WM/STM are akin to building up a final statue by adding pieces. So, OP evinces a spatiotemporal generalization of negative space vs. positive space.

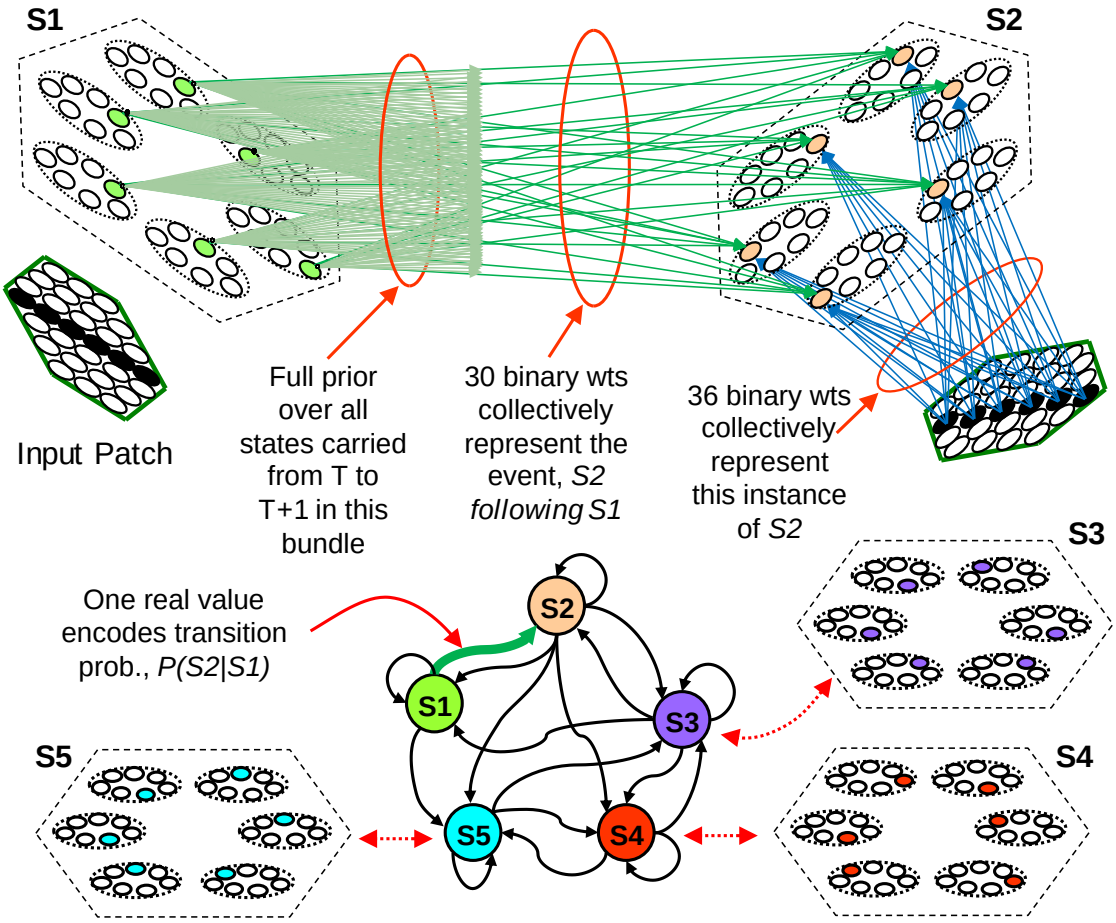
8. SDR GENERALIZES GRAPHICAL PROBABILISTIC MODELS (GPMS) AND PROVIDES TRACTABLE UNSUPERVISED STRUCTURE LEARNING:

Graphical probability models (GPMs), which include HMMs, factor graphs, dynamic Bayes nets (DBNs), etc., have been an important approach in ML. The core principle of GPMs is that while typical domains of interest may have tens/hundreds of thousands of observed and hidden variables, any given variable typically only *directly* affects or is affected by a small fraction of the other variables, which allows the full joint probability density (JPD) function to be factored into a product of much lower-dimensional JPDs. This generally exponentially reduces the number of parameters in the model, making learning and inference far more efficient. However, prior GPMs have been cast almost exclusively in localist terms, i.e., individual latent variables are represented by single representational units, which precludes the major advantages described in the prior section. One notable exception is the "flowing statistics" model of Pitkow and colleagues although thus far, *they have only addressed inference, not learning* (Rajkumar and Pitkow 2016, Pitkow and Angelaki 2017). A second, quite limiting characteristic of many GPMs is that the latent variables relevant to a domain/task and the structural/causal relations amongst them (e.g., the directed links in a Bayes net), are assumed *a priori*, imposing constraints on what can be learned and undermining domain/task neutrality. More recently, GPMs have been proposed which automatically discover the domain's variables, e.g., (Doshi-Velez, Wingate et al. 2011), and their relations. However, the learned variables (factors) are still represented in localist fashion in such models, which will therefore be subject to the suboptimalities associated with localism.

Figure 3 sketches the watershed difference between SDR-based and the mainstream localist GPMs. It shows that not only are the *states* of the latent variables themselves represented differently in localism vs. SDR, but so are the *relationships* amongst the states (or more generally amongst the latent variables themselves). In the traditional HMM concept (bottom center), the conditional probability relation between two states, e.g., p(S2|S1), is represented by a single real-valued parameter. That parameter, along with all the other relationship parameters, including those from the input level (features) to the latent variables are learned incrementally. Crucially, large numbers of instances of *transitions between whole states* must be observed in order for the model, in particular, the transition probability matrix, to approximate the true probabilities sufficiently closely.

This differs fundamentally from the learning scenario in the case of an SDR-based model. Around the outside of the Figure 3, we show hypothetical SDR codes representing states S1 to S5. Note that there is only one SDR coding field (mac) here, which is recurrent. At top, we show the 30 binary weights (green) that will have been increased *en masse* on a *single* occurrence of S2 following S1, i.e., single-trial learning [images are tilted to unclutter the horizontal (H) connections (green), which are in fact, recurrent]. Thus, these 30 weights cannot be viewed as encoding the conditional probability, p(S2|S1), as in the traditional (localist) HMM. To understand the probabilistic interpretation of this bundle, or "*synapsemble*" (Buzsaki), of 30 weights, consider the following. If one of the six green units representing S1 were not active, i.e., if some other cell won in one of the six CMs, then, as described in earlier sections, S1 would have lower likelihood/probability, even though it still might be maximal. In this case, only 25 of the 30 green weights would be active. Thus, when the CSA executes on the next time step, the H-summations to the six units representing S2 [ϕ(S2), orange] would be lower, thus increasing the chances, in each of the Q CMs, of units other than the orange unit winning. Thus, the probability of ϕ(S2) as a whole becoming active is lowered. And, this would be true even if the bottom-up input on the

next time step was the same as in the original learning instance, due to the CSA's multiplicative combination of U and H signals. Thus, even though there may have been only one instance of S2



following S1 in this model's training experience, we can interpret the synapsemble from S1 to S2 probabilistically. This is a quite abbreviated discussion of this concept. Many future research directions can be envisioned to demonstrate/test it and to relate to existing formalisms.

Figure 3: Relationship of Sparsey's representation of latent variables and their relationships to their representation in a traditional, i.e., localist, GPM, in particular, a Hidden Markov model (HMM). In the Sparsey version, there is a single recurrent SDR field (dashed hexagon) comprised of Q=6 CMs, each with K=7 units, and we assume full recurrent (i.e., horizontal, H) connectivity of L2. Each state is represented by a unique SDR code (around perimeter, color-keyed to corresponding localist representations in the HMM state model at center). The relationship of the conditional probability, p(S2|S1), as represented in the traditional HMM, to the probabilistic information encoded in the bundle, or *synapsemble*, of 30 weights from the SDR representing S1 [green units, ϕ(S1)] to that representing S2 [orange, ϕ(S2)] is described in the text.

9. AFTERWORD

These are only a sample of my research questions and initiatives. Many more novel aspects and capabilities of Sparsey can be found on my website.

10. REFERENCES

Aljundi, R., F. Babiloni, M. Elhoseiny, M. Rohrbach and T. Tuytelaars (2017) "Memory Aware Synapses: Learning what (not) to forget." ArXiv e-prints **1711**.

Curto, C., V. Itskov, K. Morrison, Z. Roth and J. L. Walker (2013). "Combinatorial Neural Codes from a Mathematical Coding Theory Perspective." Neural Comp **25**(7): 1891-1925.

Dasgupta, S., C. F. Stevens and S. Navlakha (2017). "A neural algorithm for a fundamental computing problem." Science **358**(6364): 793-796.

Doshi-Velez, F., D. Wingate, J. B. Tenenbaum and N. Roy (2011). Infinite Dynamic Bayesian Networks. 28th International Conference on Machine Learning, Bellevue, WA, USA.

Fries, P. (2015). "Rhythms for Cognition: Communication through Coherence." Neuron **88**(1): 220-235.

Fries, P., D. Nikolic and W. Singer (2007). "The gamma cycle." Trends in Neurosciences **30**(7): 309-316.

Fusi, S., E. K. Miller and M. Rigotti (2016). "Why neurons mix: high dimensionality for higher cognition." Current Opinion in Neurobiology **37**: 66-74.

Hubel, D. H. and T. N. Wiesel (1962). "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex." J Physiol **160**: 106-154.

Kirkpatrick, J., R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, D. Hassabis, C. Clopath, D. Kumaran and R. Hadsell (2017). "Overcoming catastrophic forgetting in neural networks." PNAS **114**(13): 3521-3526.

Kraska, T., A. Beutel, E. H. Chi, J. Dean and N. Polyzotis (2017) "The Case for Learned Index Structures." ArXiv e-prints **1712**.

Nandy, Anirvan S., Tatyana O. Sharpee, John H. Reynolds and Jude F. Mitchell (2013). "The Fine Structure of Shape Tuning in Area V4." Neuron **78**(6): 1102-1115.

Pitkow, X. and D. E. Angelaki (2017). "Inference in the Brain: Statistics Flowing in Redundant Population Codes." Neuron **94**(5): 943-953.

Rajkumar, V. and X. Pitkow (2016). Inference by Reparameterization in Neural Population Codes.

Rinkus, G. (2010). "A cortical sparse distributed coding model linking mini- and macrocolumn-scale functionality." Frontiers in Neuroanatomy **4**.

Rust, N. C. and J. J. DiCarlo (2010). "Selectivity and Tolerance ("Invariance") Both Increase as Visual Information Propagates from Cortical Area V4 to IT." The Journal of Neuroscience **30**(39): 12978-12995.

Warden, M. R. and E. K. Miller (2007). "The Representation of Multiple Objects in Prefrontal Neuronal Delay Activity." Cereb. Cortex **17**(suppl_1): i41-50.

Yamins, D. L. K. and J. J. DiCarlo (2016). "Eight open questions in the computational modeling of higher sensory cortex." Current Opinion in Neurobiology **37**: 114-120.

11. ENDNOTES

[i] SDR is an orthogonal concept to "sparse coding" (Olshausen & Field). Sparse coding says that because natural input spaces are generated by strongly structurally constrained worlds, such an input space can be described by a small lexicon of features (compared to the number of possible features defined on the space) and that any single input can be described by a small number of features from that lexicon. Sparse coding makes no particular statement about how such features can/should be physically represented by representational units, i.e., neurons, and in most treatments, the representation has defaulted to being localist (each feature represented by one unit). In SDR, not only is it the case that whole inputs are represented by a *set* of units, but in general, individual features (basis elements, principal components) will also be represented by (smaller) *sets* of units. The importance of this difference becomes apparent in considering the process of learning features