

1 Random Sketching, Clustering, and Short-Term 2 Memory in Spiking Neural Networks

3 **Yael Hitron**

4 Weizmann Institute of Science, Israel

5 yael.hitron@weizmann.ac.il

6 **Nancy Lynch**

7 MIT, USA

8 lynch@csail.mit.edu

9 **Cameron Musco**

10 University of Massachusetts Amherst, USA

11 cmusco@cs.umass.edu

12 **Merav Parter**

13 Weizmann Institute of Science, Israel

14 merav.parter@weizmann.ac.il

15 — Abstract —

16 We study input compression in a biologically inspired model of neural computation. We demon-
17 strate that a network consisting of a random projection step (implemented via random synaptic
18 connectivity) followed by a sparsification step (implemented via winner-take-all competition) can
19 reduce well-separated high-dimensional input vectors to well-separated low-dimensional vectors. By
20 augmenting our network with a third module, we can efficiently map each input (along with any
21 small perturbations of the input) to a unique *representative neuron*, solving a neural clustering
22 problem.

23 Both the size of our network and its processing time, i.e., the time it takes the network to
24 compute the compressed output given a presented input, are independent of the (potentially large)
25 dimension of the input patterns and depend only on the number of distinct inputs that the network
26 must encode and the pairwise relative Hamming distance between these inputs. The first two steps
27 of our construction mirror known biological networks, for example, in the fruit fly olfactory system
28 [9, 29, 17]. Our analysis helps provide a theoretical understanding of these networks and lay a
29 foundation for how random compression and input memorization may be implemented in biological
30 neural networks.

31 Technically, a contribution in our network design is the implementation of a *short-term* memory.
32 Our network can be given a desired *memory time* t_m as an input parameter and satisfies the following
33 with high probability: any pattern presented several times within a time window of t_m rounds will
34 be mapped to a single representative output neuron. However, a pattern not presented for $c \cdot t_m$
35 rounds for some constant $c > 1$ will be “forgotten”, and its representative output neuron will be
36 released, to accommodate newly introduced patterns.

37 **2012 ACM Subject Classification** Theory of computation → Design and analysis of algorithms

38 **Keywords and phrases** biological distributed computing, spiking neural networks, compressed
39 sensing, clustering, random projection, dimensionality reduction, winner-take-all

40 **Digital Object Identifier** 10.4230/LIPIcs.CVIT.2016.23



© Y.Hitron, N.Lynch, C.Musco and M.Parter;
licensed under Creative Commons License CC-BY
42nd Conference on Very Important Topics (CVIT 2016).

Editors: John Q. Open and Joan R. Access; Article No. 23; pp. 23:1–23:32



Leibniz International Proceedings in Informatics

LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

41 **1** Introduction

42 In this work we study brain-like networks that receive potentially complex and high-
 43 dimensional inputs (e.g., from sensory neurons representing odors, faces, or sounds) and
 44 encode these inputs in a very compressed way. Specifically, we consider networks with n
 45 input neurons and k output neurons, where n may be very large. When presented with up
 46 to k sufficiently different but otherwise arbitrary input patterns, the goal of the network is
 47 to represent the inputs in such a way that they can be recognized when presented again:
 48 each input should be uniquely mapped to a single *representative output neuron* that fires if
 49 that input pattern is reintroduced. Further, any small perturbations of a presented input
 50 should be recognized by the same representative neuron. We call the above problem the
 51 *neural clustering problem*.

52 Clustering, input memorization, and compression are fundamental problems in biological
 53 neural networks. Our work is also inspired by the important *novelty detection* problem
 54 [25, 41]. Novelty detection requires detecting inputs that differ significantly from previously
 55 seen inputs. It is easy to see that this problem can be solved with a neural clustering network,
 56 in which all sufficiently far inputs are mapped to different representative neurons and all
 57 sufficiently close inputs are mapped to the same neuron. A novel input is detected whenever
 58 a new representative neuron is assigned. The novelty detection problem has been considered
 59 recently in the fruit fly olfactory system [16], where it is believed to be solved using a random
 60 projection based method. The high level structure of this method closely resembles the initial
 61 stages of our clustering algorithm, and we see a major contribution of our work as providing
 62 a theoretical understanding of how random projection can be implemented in biologically
 63 inspired neural networks. For further discussion about the connection to fruit fly novelty
 64 detection see Section 1.2.

65 **1.1** Our Results

66 We study the neural clustering problem in a biologically inspired model of *stochastic spiking*
 67 *neural networks* (stochastic SNNs), which was previously defined in [33, 34, 35]. In these
 68 networks, computation proceeds in discrete rounds with each neuron either firing (spiking) in
 69 a round or remaining silent. Each neuron spikes randomly, with probability determined by
 70 its membrane potential. This potential is induced by spikes from neighboring neurons, which
 71 can have either an excitatory or inhibitory effect (increasing or decreasing the potential). In
 72 general, the input to an SNN is a stream of binary vectors, corresponding to spikes of the
 73 input neurons. In our setting we will consider a single binary vector as the input pattern
 74 and assume that each input vector is presented for a certain number of consecutive rounds
 75 before changing. This allows the network time to stabilize to the correct output associated
 76 with the given input.

77 We demonstrate that clustering can be solved efficiently in these networks, where the cost
 78 is measured by (i) the number of *auxiliary neurons*, besides the input and output neurons,
 79 that are required to solve the clustering task and (ii) the number of rounds required to
 80 converge to the correct output for a given input, which corresponds to the number of rounds
 81 for which the input must be presented for before moving to the next input.

82 In the clustering problem, we consider a (potentially large) set of n -length patterns that
 83 are clustered around k base patterns. It is then required to map all patterns in the same
 84 cluster to a unique output in $[k]$.

85
 86 **Clustering with Output Reassignment.** We also want our network to be reusable,

with a *memory duration* t_m that is given as an input parameter. Instead of considering a single infinite input stream with at most k distinct patterns (or clusters of patterns), our memory module allows one having many distinct patterns, as long as their presentation times are sufficiently spaced out. That is, in any window of $\Theta(t_m)$ rounds, the network is presented at most k distinct patterns. To handle distinct patterns in each $\Theta(t_m)$ -round window, the network must *forget* patterns that have not been introduced for a while and release their allocated outputs so that they can be assigned to new inputs. Specifically, for some fixed constant c , our network remembers a pattern (its cluster) for at least t_m rounds and at most $c \cdot t_m$ round. The output of any pattern not introduced for $c \cdot t_m$ rounds is released with high probability and can be reassigned to represent another input.

The Neural Clustering Problem. We now formally define the neural clustering problem, which is parametrized by several parameters: the input dimension n , the number of distinct input patterns k , the memory duration t_m , a bound on the relative distance of input patterns Δ , and the allowed failure probability δ . We require that every pattern introduced as input, remains the input pattern for at least $t_p = \text{poly}(k, 1/\Delta, \log(1/\delta))$ (i.e., independent of n) consecutive rounds. The t_p parameter is the processing time or mapping time, i.e., the time it takes for the network to converge to the output neuron. Throughout, we will assume that all patterns have p non-zero entries. We conjecture that this assumption can be easily removed however keep it to simplify our arguments.

Define the *relative Hamming distance* between two inputs $\bar{X}_i, \bar{X}_j \in \{0, 1\}^n$ to be:

$$\mathcal{RD}(\bar{X}_i, \bar{X}_j) = \frac{\|\bar{X}_i - \bar{X}_j\|_1}{\max\{\|\bar{X}_i\|_1, \|\bar{X}_j\|_1\}}.$$

In the basic clustering problem, the network is introduced to a possibly large number of distinct patterns that are *clustered* around k -centers. That is, in every window of t_m rounds, the patterns introduced are clustered around a base-set of k patterns $\bar{X}_1, \dots, \bar{X}_k \in \{0, 1\}^n$ such that the relative difference between each pair in the base-set is at least Δ , and any other pattern introduced is sufficiently close to one of the patterns in the base-set (with relative distance $\leq \Delta/\alpha$ for some $\alpha = \tilde{O}(1)$). In the clustering problem the network maps *similar* patterns to the same unique output q_i for $i \in [k]$ (i.e., the cluster name) and *non-similar* patterns to distinct names. Formally:

► **Definition 1** (Clustering Input Condition). *An infinite input sequence $\bar{Z}_1, \bar{Z}_2, \dots$ is a well-behaved clustering input sequence with input size n , output size k , memory duration t_m , relative distance parameter Δ , closeness parameter α , and input persistence time t_p if:*

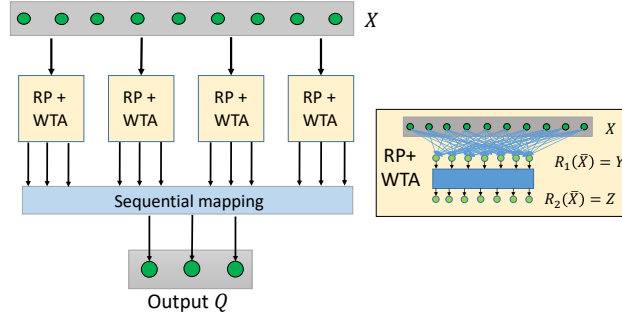
■ *For any set of t_m rounds $T = \{t, t+1, \dots, t+(t_m-1)\}$ there exist $\bar{X}_1, \bar{X}_2, \dots, \bar{X}_k \in \{0, 1\}^n$ such that $\mathcal{RD}(\bar{X}_i, \bar{X}_j) \geq \Delta$ for all $i \neq j$ and for all $i \in T$, $\mathcal{RD}(\bar{Z}_i, \bar{X}_j) \leq \Delta/\alpha$ for some $j \in [k]$.*

■ *If $\bar{Z}_i \neq \bar{Z}_{i-1}$, then $\bar{Z}_i = \bar{Z}_{i+1} = \dots = \bar{Z}_{i+t_p}$.*

► **Definition 2** (Clustering Network). *A network \mathcal{N} solves the clustering problem for input size n , output size k , memory duration t_m , relative distance parameter Δ , closeness parameter α , input duration t_p , and failure probability δ if, on a well-behaved input sequence for the same parameters (Definition 1), on any fixed window of t_m rounds, with probability $\geq 1 - \delta$:*

■ *Each input pattern \bar{Z} is mapped to some output q_j for $j \in [k]$. That is, whenever the input changes to \bar{Z} round i (so $\bar{Z}_{i-1} \neq \bar{Z}$ but $\bar{Z}_i = \bar{Z}$), there is a unique output neuron q_j that fires at round $i + t_p$ and continues to fire as long as the input remains fixed to \bar{Z} .*

■ *Any pair of far patterns \bar{Z}, \bar{Z}' with $\mathcal{RD}(\bar{Z}, \bar{Z}') \geq \Delta$ introduced within the t_m time window are mapped to different outputs.*



■ **Figure 1** High level illustration of the clustering network. Right: The input pattern $\bar{X} \in \{0, 1\}^n$ is mapped to an intermediate sparser vector in two steps: random projection and WTA sparsification. Left: In the clustering network, the input \bar{X} is mapped by applying $O(\log(k/\delta))$ parallel repetitions of the random projection + WTA mapping. As a result, \bar{X} is mapped to a vector \bar{Z} with $O(\frac{\log(k/\delta)}{\Delta})$ neurons. This vector is mapped to the output unit vector in $\{0, 1\}^k$ via a sequential mapping module.

131 ■ Any pair of closed-patterns \bar{Z}, \bar{Z}' with $\mathcal{RD}(\bar{Z}, \bar{Z}') \leq \Delta/\alpha$ introduced within the same t_m
 132 time window will be mapped to the same output neuron.

133 Our goal is to design a clustering network that uses small number of auxiliary neurons and
 134 requires small input persistence time t_p . We show the following theorem.

135 ▶ **Theorem 3.** For any parameters n, k, t_m, δ and Δ , there is a network \mathcal{N} with
 136 $O\left(\frac{\log(1/\Delta)^3 \log(t_m/\delta) \log(1/\delta)}{\Delta^{3/2}}\right)$ auxiliary neurons that solves the clustering problem with these
 137 parameters, input persistence time $t_p = O\left(\frac{\log(1/\Delta)^2 \log(t_m/\delta)}{\Delta}\right)$ and closeness parameter
 138 $\alpha = O(\log(1/\Delta)^4)$.

139 Note that the number of auxiliary neurons and the convergence time of Theorem 3 are
 140 independent of the input dimension n , which may potentially be very large. The spiking
 141 neural network construction that achieves Theorem 3 involves in three steps. The first two
 142 steps reduce the input from n neurons to $m \ll n$ neurons, while approximately preserving the
 143 relative distances between inputs. These steps use a biologically inspired construction that
 144 mirrors circuits seen, for example, in the fruit fly olfactory system [9, 29, 17]. In particular
 145 the first step maps the input to a set of intermediate neurons via random projection, and
 146 the second step sparsifies the outputs of these intermediate neurons to yield a sparse code
 147 representing the input. The final *sequential mapping* step then solves the clustering problem
 148 given these m intermediate neurons as inputs, avoiding the high cost of directly solving the
 149 problem on the n -dimensional input. See Figure 1 for an illustration.

150 1.2 Comparison to Previous Work

151 1.2.1 Broader Agenda: Algorithmic Theory for Brain Networks

152 Understanding how the brain works, as a computational device, is a central challenge of
 153 modern neuroscience and artificial intelligence. Different research communities tackle this
 154 problem in different ways, ranging from studies that examine neural network structure as a
 155 clue to computational function [43, 3], to functional imaging that studies neural activation
 156 patterns [40, 31], to theoretical work on simplified models of neural computation [23, 36],
 157 to the engineering of complex neural-inspired machine learning architectures [21, 27]. This

158 paper joins a recent line of work [44, 37, 45, 38, 33, 30, 46, 34, 28, 32, 39, 10, 22] that
 159 approach this problem using techniques from *distributed computing theory and other branches*
 160 *of theoretical computer science*. The ultimate goal of this research direction is to develop
 161 an *algorithmic theory for brain networks*, based on stochastic graph-based neural network
 162 models. To understand neural behavior from a theory of computing point of view, we design
 163 networks to solve abstract problems that are inspired by tasks that seem to be solved in
 164 actual brains. We believe that the rigorous analysis of such networks in terms of static costs
 165 (e.g., the number of neurons), and dynamic costs (e.g., the time to converge to a solution)
 166 will lead to a better understanding of how these tasks may be performed in biological neural
 167 networks.

168 1.2.2 Connections to Sparse Recovery

169 Our work is closely related to sparse recovery (compressed sensing), where the goal is to map
 170 high-dimensional but sparse vectors (with dimension n and $s \ll n$ nonzero entries) into a
 171 much lower dimensional space, such that the vectors can be uniquely identified and efficiently
 172 recovered [19]. We can see that this goal is essentially identical to that of our first two network
 173 layers, before the sequential mapping step. Two different s -sparse binary vectors have relative
 174 hamming distance $\geq 1/s$. Additionally there are $k = O(n^s)$ s -sparse binary vectors in n
 175 dimensions. Thus, as a Corollary of Lemma 12, our first two layers can uniquely compress
 176 all such vectors with high probability into dimension $m = \tilde{O}\left(\frac{\log k}{\Delta^{1/2}}\right) = O(s^{3/2} \log n)$.

177 It is known that optimal sparse recovery reducing the dimension to $O(s \log n)$ can be
 178 achieved using random projections [13]. However, unlike in our setting, these random
 179 projections have real valued outputs, which cannot be directly represented by binary spiking
 180 neurons. The case when output of the random projection is thresholded to be a binary value
 181 has been studied extensively, under the name ‘one-bit compressed sensing’ [6]. In this setting,
 182 it is known that dimension $\tilde{\Theta}(s^2 \log n)$ can be achieved and is required for general sparse
 183 recovery [1]. If the input vectors are restricted to be binary (as in our case), dimension
 184 $O(s^{3/2} \log n)$ is possible [24, 1]. Our results match this bound up to logarithmic factors.

185 1.2.3 Connections to Fruit Fly Novelty Detection via Bloom Filters

186 Recently, Dasgupta et al. [16] demonstrated that the fruit fly olfactory circuit implements a
 187 variant of a classic Bloom filter [5] to assess the novelty of odors. A bloom filter is a data
 188 structure that maintains a set of items, allowing for membership queries, with the possibility
 189 of occasional false positives. The filter has m bits and r random hash functions mapping
 190 the input space to integers in $1, \dots, m$. When an item is inserted, it is hashed using these r
 191 functions and the bits corresponding to the hashed values are set to 1. A membership query
 192 is answered by hashing the input in question and checking that all r bits corresponding to
 193 its hashed values are set to 1.

194 Such a filter can be used to implement novelty detection – a novel pattern is detected
 195 whenever an insertion operation sets a new bit to 1 or a membership query returns false.
 196 Dasgupta et al. [16] demonstrate that a such a scheme is used in the fly olfactory circuit.
 197 The hashing step consists of a random projection followed by winner-take-all sparsification,
 198 which maps each input into a r -sparse binary vector. The r entries of this vector represent
 199 the r hash function outputs. This step closely resembles the first two layers of our clustering
 200 network.

201 Our third layer operates differently than a bloom filter, associating each sparsified
 202 intermediate vector to with unique output via sequential mapping rather than simply marking

203 the bits corresponding to its entries. However, it can implement the same functionality
 204 (and correspondingly can implement novelty detection). Specifically, to implement insertion
 205 and deletion operations we can make the following modifications to the sequential mapping
 206 sub-network:

- 207 ■ The input layer contains an extra neuron that is set to 1 if the operation is insertion, and
 208 0 if the operation is a membership query.
- 209 ■ In the sequential mapping step the output layer fires only if this extra neuron fires. In this
 210 way, new outputs will only be mapped during insertion operations and not membership
 211 queries.
- 212 ■ For query operations, we add an output neuron that fires only if there exists an index
 213 $j \in [k]$ for which many memory modules m_{ji} , as well as an association neuron a_{ji} fire.
- 214 ■ Novelty detection can be implemented via an additional output neuron that responds
 215 when an insertion causes a new output to be mapped or when a query operation returns
 216 false.

217 **2 Computational Model and Preliminaries**

218 We start by defining our model of stochastic spiking neural networks.

219 **Network Structure.** A *Spiking Neural Network* (SNN) $\mathcal{N} = \langle X, Q, A, w, b \rangle$ consists of
 220 n input neurons $X = \{x_1, \dots, x_n\}$, m output neurons $Q = \{q_1, \dots, q_m\}$, and ℓ auxiliary
 221 neurons $A = \{a_1, \dots, a_\ell\}$. The directed, weighted synaptic connections between X , Q , and A
 222 are described by the weight function $w : [X \cup Q \cup A] \times [X \cup Q \cup A] \rightarrow \mathbb{R}$. A weight $w(u, v) = 0$
 223 indicates that a connection is not present between neurons u and v . Finally, for any neuron
 224 v , $\beta(v) \in \mathbb{R}_{\geq 0}$ is the activation bias – as we will see, roughly, v 's membrane potential must
 225 reach $\beta(v)$ for a spike to occur with good probability.

226 The in-degree of every input neuron x_i is zero. That is, $w(u, x) = 0$ for all $u \in [X \cup Q \cup A]$
 227 and $x \in X$. Additionally, each neuron is either *inhibitory* or *excitatory*: if v is inhibitory,
 228 then $w(v, u) \leq 0$ for every u , and if v is excitatory, then $w(v, u) \geq 0$ for every u .

229 **Neuron Chains.** In our implementation, we make use of *chains* of neurons to create a
 230 delay in a response. For a neuron u , and integer ℓ , let $C_\ell(u)$ be a directed path of length ℓ
 231 starting with u . All neurons on the chain are excitatory. We then say that a chain $C_\ell(u)$ is
 232 connected to v if each neuron $w \in C_\ell(u)$ has an outgoing edge to v .

233 **The SNN Model.** An SNN evolves in discrete, synchronous rounds as a Markov chain.
 234 The firing probability of every neuron at time t depends on the firing status of its neighbors
 235 at time $t - 1$, via a standard sigmoid function, with details given below. For each neuron u ,
 236 and each time $t \geq 0$, let $u^t = 1$ if u fires (i.e., generates a spike) at time t . Let u^0 denote the
 237 initial firing state of the neuron. Our results will specify the initial input firing states $x_j^0 = 1$
 238 and assume that $u^0 = 0$ for all $u \in [Q \cup A]$. The firing state of each input neuron x_j in each
 239 round is the input to the network, and our results will specify to which sequences of input
 240 firing patterns they apply.

241 For each non-input neuron u and every $t \geq 1$, let $\text{pot}(u, t)$ denote the membrane potential
 242 at round t and $p(u, t)$ denote the corresponding firing probability ($\Pr[u^t = 1]$). These values
 243 are calculated as:

$$244 \quad \text{pot}(u, t) = \sum_{v \in [X \cup Q \cup A]} w_{v,u} \cdot v^{t-1} - \beta(u) \quad \text{and} \quad p(u, t) = \frac{1}{1 + e^{-\text{pot}(u,t)/\lambda}} \quad (1)$$

246 where $\lambda > 0$ is a *temperature parameter*, which determines the steepness of the sigmoid. It is
 247 easy to see that λ does not affect the computational power of the network. A network can

248 be made to work with any λ simply by scaling the synapse weights and biases appropriately.
 249 For simplicity we assume throughout that $\lambda = \frac{1}{\Omega(\log(n \cdot k \cdot \Delta \cdot t_m \cdot 1/\delta))}$, where $n, k, \delta, \Delta, t_m$ are the
 250 parameters of the clustering problem, defined in Section 1.1. Thus by (1), if $\text{pot}(u, t) \geq 1$,
 251 then $u^t = 1$ w.h.p. and if $\text{pot}(u, t) \leq -1$, $u^t = 0$ w.h.p. , where w.h.p. denotes with
 252 probability at least $1 - (1/\delta \cdot n \cdot k \cdot \Delta \cdot t_m)^{-c}$ for some constant c .

253 The remainder of the paper is devoted to proving Theorem 3. Our analysis considers the
 254 three stages of the network in sequence: random projection, sparsification, and sequential
 255 mapping to the final outputs.

256 **3 Layer 1: Random Projection**

257 The goal of this step is to reduce the input size from n input neurons to $m \ll n$ neurons while
 258 ensuring that the relative distance between any two n -length input vectors is approximately
 259 preserved after the mapping. In this way, we can solve the clustering problem working with
 260 the much smaller m neuron representation instead of the original n neuron input. While
 261 there are many ways in which distance may be preserved, we consider one in particular,
 262 based on the membrane potentials induced on the intermediate neurons by the inputs:

263 **► Definition 4** (Distance Preserving Dimensionality Reduction). Consider $\bar{X}_1, \dots, \bar{X}_k \in \{0, 1\}^n$
 264 with $\mathcal{RD}(\bar{X}_i, \bar{X}_j) \geq \Delta$ for $i \neq j$. Consider a network \mathcal{N} mapping n input neurons to m
 265 intermediate neurons, which are split into b buckets each containing m/b neurons. \mathcal{N} is
 266 distance preserving for $\bar{X}_1, \dots, \bar{X}_k$ if, for any two \bar{X}_i, \bar{X}_j , and any round t , in the large
 267 majority of buckets, the identity of the neuron that in round $t + 1$ has the highest membrane
 268 potential below a fixed threshold τ is different if \bar{X}_i is presented at round t than if \bar{X}_j were
 269 presented.¹

270 Our network satisfies Definition 4 with parameters $m = \tilde{O}\left(\frac{1}{\sqrt{\Delta}}\right)$ and $b = \tilde{O}(1)$. We
 271 implement the dimensionality reduction step via *random projection*. We note that random
 272 projection has been studied extensively as a dimensionality reduction tool in computer
 273 science, with applications in data analysis [4, 7, 12], fast linear algebraic computation [42, 11],
 274 and sparse recovery [8]. See [47] for a survey. In neuroscience, it is thought that random
 275 projection may play a key role in neural dimensionality reduction [20, 2]. Random projection
 276 for example, underlies sparse coding of inputs in the fruit fly olfactory circuit [9, 17]. Random
 277 connections have also been studied in theoretical models for memory formation, in which
 278 inputs are mapped to representative output neurons [45, 38, 28].

279 We start with describing the construction and then analyzing its properties. The main
 280 outputs of this section are Corollaries 9 and 11 which show that, with high probability, the
 281 identities of the neurons with maximum membrane potential below some threshold τ in each
 282 bucket of the intermediate layer share little overlap for far inputs (with relative distance $\geq \Delta$)
 283 and significant overlap for close inputs (with relative distance $\leq \Delta/\alpha$ for $\alpha = O(\log(1/\Delta)^4)$).
 284 That is, the network satisfies the distance preserving dimensionality reduction guarantee of
 285 Definition 4 for far inputs, along with an analogous guarantee for close inputs.

286 Our mapping can be understood as an example of *local sensitive hashing* [18, 15, 17]. In
 287 each bucket, the input is hashed to the identity of the maximum potential neuron below τ
 288 in that bucket. Near inputs have many hash collisions, and thus there is significant overlap in
 289 the identities of the mapped neurons. Far input have fewer collisions and thus less overlap.

¹ We formally define how the membrane potential is calculated in Section 2. ‘Large majority’ will be a constant fraction of the buckets significantly larger than $1/2$, which will be specified in our bounds.

290

291 **Layer Description.** The random projection layer consists of $m \cdot \ell$ intermediate auxili-
 292 ary neurons for $m = \Theta\left(\frac{\log(1/\Delta)}{\sqrt{\Delta}}\right)$ and $\ell = \Theta(\log(t_m/\delta))$. The layer is subdivided into ℓ
 293 buckets b_1, \dots, b_ℓ containing m neurons each. Each input neuron has an excitatory connection
 294 to each neuron in the intermediate layer with weight sampled as a Chi-squared random
 295 variable (with one degree of freedom). We denote this random weight matrix connecting
 296 the two layers by $A \in \mathbb{R}^{m \cdot \ell \times n}$. For $b \in 1, \dots, \ell$, we let $A_b \in \mathbb{R}^{m \times n}$ denote the rows of A
 297 corresponding to the intermediate neurons in bucket b . In typical applications of random
 298 projection, the entries of A are most commonly either Gaussian or Rademacher random
 299 variable. Here we use Chi-squared random variables as they are non-negative, a requirement
 300 in our setting where the outgoing edge weights from each neuron (corresponding to the
 301 entries in A) must be either all positive or all negative.

302

303 **Layer Analysis.** When the input neurons X fire with input pattern $\bar{X}_i \in \{0, 1\}^n$ at
 304 time t , by (1) the vector of membrane potentials of the intermediate neurons at time $t + 1$ is
 305 given by $A \bar{X}_i \in \mathbb{R}^{m \cdot \ell}$. Our analysis will focus on the properties of this vector of potentials,
 306 which can be viewed as a real valued compressed representation of the input \bar{X}_i . Later, we
 307 will show how these properties lead to desirable properties of the spiking patterns of the
 308 intermediate neurons.

309 For technical reasons, we will not focus on the actual largest entry of $A_b \bar{X}_i$, but on the
 310 *largest entry bounded by some fixed threshold τ* which can still be identified via a minor
 311 modification to a traditional WTA circuit. We begin with a preliminary lemma showing
 312 that a Chi-squared distribution (the distribution of each entry in $A_b \bar{X}_i$) is roughly uniform
 313 around its mean. We give a proof in Appendix A.

314 **► Lemma 5 (Chi-squared uniformity).** *Let \mathcal{D}_p be the Chi-squared distribution with p degrees*
 315 *of freedom. For any c with $1 \leq c < p^{1/2}$ there are constants c_ℓ, c_u (depending on c) such that,*
 316 *for any interval $[r_1, r_2] \subseteq [p - cp^{1/2}, p + cp^{1/2}]$, we have: $\frac{c_\ell(r_2 - r_1)}{p^{1/2}} \leq \Pr_{x \sim \mathcal{D}_p}[x \in [r_1, r_2]] \leq$*
 317 *$\frac{c_u(r_2 - r_1)}{p^{1/2}}$. That is, \mathcal{D}_p is roughly uniform on the range $[p - cp^{1/2}, p + cp^{1/2}]$.*

318 We also use the fact that the Chi-squared distribution decays far from its mean, which follows
 319 from standard sub-exponential concentration bounds.

320 **► Lemma 6 (Chi-squared decay).** *Let \mathcal{D}_p be the Chi-squared distribution with p degrees of*
 321 *freedom. For any $c \leq 1$ there is a constant c_1 (depending on c) such that:*

$$322 \Pr_{x \sim \mathcal{D}_p} \left[x \notin [p - c_1 p^{1/2}, p + c_1 p^{1/2}] \right] \leq c.$$

324 Using the near-uniform distribution property of Lemma 5, we can show that with good
 325 probability, for every compressed vector $A_b \bar{X}_i \in \mathbb{R}^m$ the gap between the two largest entries
 326 (bounded by the threshold) is $\Omega(p^{1/2}/m)$ – since there are m entries roughly uniformly
 327 distributed in a range of size $O(p^{1/2})$. This gap will be necessary for the neuron with the
 328 largest membrane potential (and hence the highest firing probability) to be reliably identified
 329 in the second sparsification layer of our network. We remark that in non-neural applications
 330 of random projection such a gap would not be necessary: the largest entry in the bucket can
 331 be typically be identified exactly.

332 The complete proof is given in Appendix A.

► Lemma 7 (Sufficient Gap). *Consider our construction with bucket size $m = c_1$. Let*
 $\bar{X} \in \{0, 1\}^n$ be any input vector with $\|\bar{X}\| = p$ for $p \geq 5$. Let $\tau = p + 2p^{1/2}$ and for any

$b \in [\ell]$ define:

$$i_{1,b}(\bar{X}) = \arg \max_{i \in [m]: [A_b \bar{X}](i) \leq \tau} [A_b \bar{X}](i) \quad \text{and} \quad i_{2,b}(\bar{X}) = \arg \max_{i \in [m] \setminus i_{1,b}(\bar{X}): [A_b \bar{X}](i) \leq \tau} [A_b \bar{X}](i),$$

333 where we set $i_{1,b}(\bar{X}), i_{2,b}(\bar{X}) = 0$ in the case that no index satisfies the constraint. For
 334 sufficiently large constants c_1, c_2 , with probability $\geq 99/100$ over the random choice of A_b ,
 335 $i_{1,b}(\bar{X}) \neq 0$, $[A_b \bar{X}](i_{1,b}(\bar{X})) \geq p$, and either $[A_b \bar{X}](i_{1,b}(\bar{X})) - [A_b \bar{X}](i_{2,b}(\bar{X})) \geq \frac{p^{1/2}}{c_2 m}$ or
 336 $i_{2,b}(\bar{X}) = 0$.

337 Along with Lemma 7 we prove that, with good probability, the neuron with the maximum
 338 potential below τ in each bucket differs for any two far inputs.

339 ► **Lemma 8** (Low Collision Probability – Far Inputs). Let $\bar{X}_1, \bar{X}_2 \in \{0, 1\}^n$ be two vectors
 340 with $\|\bar{X}_1\| = \|\bar{X}_2\| = p^2$ and $\mathcal{RD}(\bar{X}_1, \bar{X}_2) \geq \Delta$. Assume that $p \geq c$ for some sufficiently
 341 large constant c . Consider our construction with bucket size $m = \frac{c_1 \log(1/\Delta)}{\sqrt{\Delta}}$. Then for
 342 sufficiently large constants c_1, c_2 , for any $b \in [\ell]$, defining $i_{1,b}(\cdot)$ and $i_{2,b}(\cdot)$ as in Lemma 7,
 343 with probability ≥ 0.9165 :

- 344 ■ $i_{1,b}(\bar{X}_1) \neq i_{1,b}(\bar{X}_2)$.
- 345 ■ For both $j = 1, 2$: $i_{1,b}(\bar{X}_j) \neq 0$, $[A_b \bar{X}_j](i_{1,b}(\bar{X}_j)) \geq p$, and
 346 $[A_b \bar{X}_j](i_{1,b}(\bar{X}_j)) - [A_b \bar{X}_j](i_{2,b}(\bar{X}_j)) \geq \frac{p^{1/2}}{c_2 \cdot m}$ or $i_{2,b}(\bar{X}_j) = 0$.

347 See Appendix A for the complete proof of Lemma 8. Intuitively, if \bar{X}_1 and \bar{X}_2 each
 348 have Hamming weight p and relative distance Δ they differ on $\Omega(\Delta p)$ entries. If *just the*
 349 *shared entries* of these vectors fired as inputs, by Lemma 5 each intermediate neuron in
 350 the bucket of size m would have its potential distributed roughly uniformly in a range of
 351 width $O([(1 - \Delta)p]^{1/2}) = O(p^{1/2})$. On average these potentials would be spaced out by
 352 $O(p^{1/2}/m)$. By setting $m = \tilde{O}(1/\sqrt{\Delta})$ we have average spacing $\tilde{O}(\Delta^{1/2} p^{1/2})$. This is a small
 353 enough spacing, so that when we consider the $\Omega(\Delta p)$ non-shared neurons in the inputs, their
 354 contribution to the potential will be large enough to significantly reorder the potentials of
 355 the intermediate neurons, so that the neuron with maximum potential is unlikely to be the
 356 same for the two different inputs.

357 From Lemma 8 we can show that our network satisfies the distance preserving dimen-
 358 sionality reduction guarantee of Definition 4, along with the additional condition that the
 359 gap between the membrane potentials of the neurons with the largest potentials under
 360 $\tau = p + 2p^{1/2}$ is sufficiently large, so that these neurons can be distinguished reliably in the
 361 second sparsification layer:

362 ► **Corollary 9** (Overall Success – Far Inputs). For $m = O\left(\frac{\log(1/\Delta)}{\sqrt{\Delta}}\right)$, and $\ell = O(\log(t_m/\delta))$,
 363 for any window of t_m rounds, with probability $\geq 1 - \delta$, for all pairs of inputs \bar{X}_1, \bar{X}_2 presented
 364 during these rounds with $\mathcal{RD}(\bar{X}_1, \bar{X}_2) \geq \Delta$, on at least $91/100 \cdot \ell$ of the ℓ buckets, letting
 365 $\tau = p + 2p^{1/2}$ and defining $i_{1,b}(\cdot)$ and $i_{2,b}(\cdot)$ as in Lemma 7:

- 366 ■ $i_{1,b}(\bar{X}_1) \neq i_{1,b}(\bar{X}_2)$
- 367 ■ For both $j = 1, 2$: $i_{1,b}(\bar{X}_j) \neq 0$, $[A_b \bar{X}_j](i_{1,b}(\bar{X}_j)) \geq p$, and
 368 $[A_b \bar{X}_j](i_{1,b}(\bar{X}_j)) - [A_b \bar{X}_j](i_{2,b}(\bar{X}_j)) = \Omega\left(\frac{p^{1/2}}{m}\right)$ or $i_{2,b}(\bar{X}_j) = 0$.

² When \bar{X} is binary we often drop the subscript and just use $\|\bar{X}\|$ to denote the ℓ_1 norm which is equal to the number of nonzero entries, $|\text{supp}(\bar{X})|$.

369 **Proof.** By Lemma 8 and a Chernoff bound, since $\ell = \Theta(\log(t_m/\delta)) = \Theta(\log(t_m^2/\delta))$, for any
 370 fixed pair of inputs with $\mathcal{RD}(\bar{X}_1, \bar{X}_2) \geq \Delta$, the conditions hold on at least $91/100 \cdot \ell$ buckets
 371 with probability $\geq 1 - \delta/t_m^2$. The corollary follows since at most t_m inputs can be presented
 372 in t_m rounds, and so we can union bound over at most t_m^2 pairs of far inputs. ◀

373 We can give a complementary statement to Lemma 8: if \bar{X}_1 and \bar{X}_2 are close to each
 374 other, it is relatively likely that the index of the largest value of $A_b \bar{X}_1$ and $A_b \bar{X}_2$ are the
 375 same. We defer the proof to Appendix A.

376 ► **Lemma 10 (High Collision Probability – Close Inputs).** *Let $\bar{X}_1, \bar{X}_2 \in \{0, 1\}^n$ be two vectors
 377 with $\mathcal{RD}(\bar{X}_1, \bar{X}_2) \leq \Delta/\alpha$. Consider our construction with bucket size $m = \frac{c_1 \log(1/\Delta)}{\sqrt{\Delta}}$. Then
 378 for sufficiently large constants c_1, c_2 and $\alpha = O(\log(1/\Delta)^4)$, for any $b \in [\ell]$, defining $i_{1,b}(\cdot)$
 379 and $i_{2,b}(\cdot)$ as in Lemma 7, with probability ≥ 0.97 :*

- 380 ■ $i_{1,b}(\bar{X}_1) = i_{1,b}(\bar{X}_2)$.
- 381 ■ For both $j = 1, 2$: $i_{1,b}(\bar{X}_j) \neq 0$, $[A_b \bar{X}_j](i_{1,b}(\bar{X}_j)) \geq p$, and
- 382 $[A_b \bar{X}_j](i_{1,b}(\bar{X}_j)) - [A_b \bar{X}_j](i_{2,b}(\bar{X}_j)) \geq \frac{p^{1/2}}{c_2 m}$ or $i_{2,b}(\bar{X}_j) = 0$.

383 Lemma 10 yields an analogous corollary to Corollary 9, which follows via a Chernoff
 384 bound and a union bound over at most t_m^2 pairs of close inputs that may be presented over
 385 t_m rounds.

386 ► **Corollary 11 (Overall Success – Close Inputs).** *For $m = O\left(\frac{\log(1/\Delta)}{\sqrt{\Delta}}\right)$, $\ell = O(\log(t_m/\delta))$,
 387 and $\alpha = O(\log(1/\Delta)^4)$, for any window of t_m rounds, with probability $\geq 1 - \delta$, for all pairs of
 388 inputs \bar{X}_1, \bar{X}_2 presented during these rounds with $\mathcal{RD}(\bar{X}_1, \bar{X}_2) \leq \Delta/\alpha$, on at least $96/100 \cdot \ell$
 389 of the ℓ buckets, letting $\tau = p + 2p^{1/2}$ and defining $i_{1,b}(\cdot)$ and $i_{2,b}(\cdot)$ as in Lemma 7:*

- 390 ■ $i_{1,b}(\bar{X}_1) = i_{1,b}(\bar{X}_2)$
- 391 ■ For both $j = 1, 2$: $i_{1,b}(\bar{X}_j) \neq 0$, $[A_b \bar{X}_j](i_{1,b}(\bar{X}_j)) \geq p$, and
- 392 $[A_b \bar{X}_j](i_{1,b}(\bar{X}_j)) - [A_b \bar{X}_j](i_{2,b}(\bar{X}_j)) = \Omega\left(\frac{p^{1/2}}{m}\right)$ or $i_{2,b}(\bar{X}_j) = 0$.

393 4 Layer 2: Sparsification via Winner Takes All

394 Corollaries 9 and 11 show that the random projection step preserves significant information
 395 about input distance, encoded in the membrane potentials of the intermediate neurons, which
 396 correspond to the entries of $A \bar{X}$ when the network is given input \bar{X} . These membrane
 397 potentials cause the intermediate neurons to fire randomly, as Bernoulli processes with
 398 different rates. The goal of our second layer is to convert this random behavior to a uniquely
 399 identifying sparse code for each input. We achieve this through a winner-takes-all (WTA)
 400 based sparsification process, which is thought to play a major role in neural computation
 401 [26, 14, 37]. A separate winner-take-all instance is applied to each bucket, ‘selecting’ the
 402 neuron with the highest membrane potential below τ by inducing its corresponding neuron
 403 in the sparsification layer to fire with high probability while all other neurons in the bucket
 404 do not fire. Let $\bar{Y} \in \mathbb{R}^m$ denote the vector of membrane potentials of a single bucket of the
 405 intermediate layer: $\bar{Y} = A_b \bar{X}$. Our WTA layer maps each \bar{Y} into a binary unit-vector \bar{Z} of
 406 the same length, in which the only firing neuron corresponds to the neuron with the largest
 407 potential in \bar{Y} that is bounded by the threshold parameter τ . As explained in Section 3, the
 408 random projection step produces $\ell = O(\log(t_m/\delta))$ random compressed vectors, one for each
 409 of the ℓ buckets. Each such copy is an input to an independent WTA circuit and thus, in
 410 this section, we focus on our construction restricted to just a single bucket, bearing in mind
 411 that in fact our network consists of ℓ repetitions of this module.

412 The first part of the WTA circuit is devoted to *reading*: the circuit collects firing statistics
 413 for a period of $T = \tilde{O}(m^2)$ rounds to obtain a good estimate of the neuron in the bucket
 414 that 1) has potential $\leq \tau$ and 2) has the largest firing rate. This neuron corresponds to the
 415 neuron with the highest potential in \bar{Y} bounded by τ . This is done by augmenting each
 416 neuron i in the bucket with a directed chain H_i of neurons of length T . The j^{th} neuron in
 417 the chain triggers the firing of the $(j+1)^{\text{th}}$ neuron with high probability. As a result, after T
 418 rounds, the number of firing neurons in the chain H_i is equal to the number of times i fired
 419 within the last T rounds, with high probability. We thus refer to this H_i chain as the *history*
 420 chain of the i^{th} neuron in the bucket. The second part of the circuit first excludes all neurons
 421 with potential $\geq \tau$ and then applies a standard WTA circuit to pick the neuron remaining
 422 that fires the most in this T -length time interval. See Fig. 2 for an illustration of the overall
 423 clustering network and the WTA module. The main result of this section is as follows.

424 ► **Lemma 12.** *For every pair of input patterns \bar{X}_i, \bar{X}_j presented over a period of t_m rounds,*
 425 *with probability at least $1 - \delta$ the following hold:*

- 426 (I) *If $\mathcal{RD}(\bar{X}_i, \bar{X}_j) \geq \Delta$, then $\text{supp}(\bar{Z}_i) \setminus \text{supp}(\bar{Z}_j) \geq 0.9 \cdot \ell$.*
 427 (II) *If $\mathcal{RD}(\bar{X}_i, \bar{X}_j) \leq \Delta/\alpha$, then $\text{supp}(\bar{Z}_i) \cap \text{supp}(\bar{Z}_j) \geq 0.9 \cdot \ell$.*

428 We first give a detailed description of the specification step via WTA (see Figure 2). We
 429 focus on a single bucket, bearing in mind that in fact our network consists of ℓ repetitions of
 430 this module.

431

432 **Reading via History Chain.** Every neuron $i \in \{1, \dots, m\}$ in the bucket is connec-
 433 ted to a chain H_i of length $T = \Theta(\log(1/\delta) \cdot m^2)$ of neurons where the j^{th} neuron in this
 434 chain fires in round t with high probability iff its incoming neighbor on that chain fires in
 435 round $t - 1$. This is done by setting the bias value of each neuron to 1 and the edge weights
 436 to be $1/2$. As a result we get that the number of firing neurons in this chain equals to the
 437 number of times i fires within the last T rounds with high probability.

438

439 **Omitting the Neurons Exceeding the Threshold Value.** For every neuron $i \in$
 440 $\{1, \dots, m\}$ we introduce an inhibitor copy r_i that has the same incoming weights as i
 441 and therefore also has the same potential. We set the bias of r_i such that with high probabili-
 442 ty it fires iff its potential exceeds the threshold value τ . We then connect r_i to all neurons
 443 in the chain H_i with large negative weight. As a result, if the potential of neuron i exceeds τ
 444 with high probability all neurons in H_i will not fire.

445

446 **Selecting the Maximum Firing Rate with Pairwise Comparisons.** For every ordered
 447 pair of neurons $i, j \in [1, m]$, we have a designated (threshold gate) neuron $y_{i,j}$ that fires iff
 448 the i^{th} neuron in the bucket fires more than the j^{th} neuron within the last T rounds. To
 449 accomplish this, each of the neurons in the chain H_i (respectively, H_j) is connected to $y_{i,j}$
 450 with a positive (respectively, negative) edge weight of ± 1 . Hence, the total weighted sum
 451 incoming to $y_{i,j}$ is exactly the difference between $R(i)$ and $R(j)$ where $\bar{R}(i), \bar{R}(j)$ are the
 452 number of times that the i^{th} and j^{th} neurons fired in the last T rounds. We set the bias of
 453 $y_{i,j}$ such that it fires with high probability iff $\bar{R}(i) - \bar{R}(j) \geq 1$. The i^{th} output neuron in
 454 \bar{Z} computes the AND-gate of the threshold-gates $y_{i,1}, \dots, y_{i,m}$. That is, z_i fires in round t
 455 only if every $y_{i,1}, \dots, y_{i,m}$ fired in round $t - 1$. The AND-gate can be implemented by set-
 456 ting the incoming edge weight from each $y_{i,j}$ to z_i to be $1/m$, and the bias of \bar{Z}_i to $1 - 1/(2m)$.

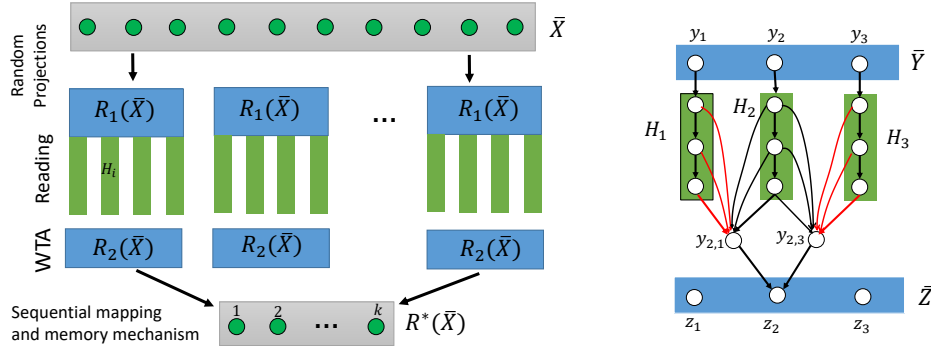
457

458 **Analysis.** The requirement from the WTA module is that the firing frequency vector
 459 \bar{R} has its largest entry in the same position as the largest entry of \bar{Y} that is $\leq \tau$. If this is the

460 case, the WTA circuit indeed selects the neuron corresponding to the largest firing rate $\leq \tau$,
 461 and the only entry in the support of \bar{Z} is the one corresponding to this entry. For the largest
 462 entry in \bar{R} to reflect the largest entry in $\bar{Y} \leq \tau$ with probability $\geq 1 - \delta$, the gap between the
 463 largest and second largest firing rates must be $\Omega\left(\sqrt{\log(1/\delta)/T}\right)$. Using the gap condition of
 464 Corollary 9 we will show that this gap is $\Omega(1/m)$, letting us set $T = O(\log(1/\delta) \cdot m^2)$. The
 465 desired gap is achieved in a large fraction of the buckets, this implies that the WTA picks
 466 the maximal entry in most of the buckets as well.

467 \triangleright **Claim 13.** Let \bar{Y} be a vector with $i = \arg \max_{j: \bar{Y}(j) \leq \tau} \bar{Y}(j)$ and $\bar{Y}(i) - \bar{Y}(j) = \Omega(p^{1/2}/m)$
 468 for every³ $j \neq i$ with $\bar{Y}(j) \leq \tau$. Then in the output vector \bar{Z} , $\bar{Z}(i) = 1$ and $\bar{Z}(j) = 0$ for
 469 every $j \neq i$ with probability at least 99/100. If \bar{Y} is first introduced in round t , the desired
 470 output vector \bar{Z} fires in round $t + T + 2$ w.h.p.

471 The proof of Claim 13 and the complete proof of Lemma 12 is given in Appendix B.

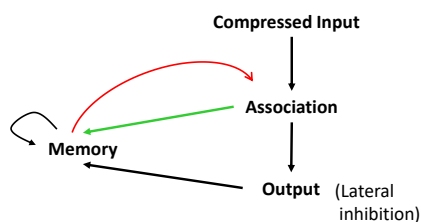


■ **Figure 2** Left: Overall network description, the input pattern \bar{X} is mapped to unique output neuron in $[1, k]$ via three main steps. Right: Description of the WTA circuit. For clarity we only show the connections for the second output neuron, but same holds for all k output neurons. Every input neuron i in \bar{Y} is connected to a history chain H_i of length T that is used to collect firing statistics. For each pair of input neurons i, j , there is a threshold gate $q_{i,j}$ that fires only if i fired at least $T/2m$ more times than j within T rounds. Each history neuron in H_i, H_j is connected with weight 1 (respectively -1) to $q_{i,j}$ and the bias of $q_{i,j}$ is $T/2m$. Finally, each output neuron q_i computes the AND gate of $q_{i,1}, \dots, q_{i,m}$, i.e., fires only if all these gates fire in the previous round. As a result a winner q_i is selected only if $y_i - y_j = \Omega(1/m)$ for every $j \neq i$.

472 5 Layer 3: Sequential Mapping

473 We conclude by discussing the final sequential mapping layer of our network, which maps
 474 the *binary* patterns \bar{Z}_i of length $r = O(\ell \cdot m)$ to a single output neuron. The inputs to the
 475 third layer are the r neurons $Z = \{z_1, z_2, \dots, z_r\}$ and its outputs are the k output neurons
 476 $Q = \{q_1, q_2, \dots, q_k\}$. The r -length patterns will be mapped to their unique output neuron in
 477 a sequential manner, where at each given round, a newly introduced pattern will be mapped
 478 to the available output with the smallest index. The mapping will satisfy the following
 479 properties: (1) patterns \bar{Z}_i, \bar{Z}_j that correspond to *far* input patterns \bar{X}_i, \bar{X}_j respectively will

³ This required gap is based on Lemma 7/Corollary 9.



■ **Figure 3** Schematic description.

480 be mapped to distinct outputs, (2) patterns \bar{Z}_i, \bar{Z}_j that correspond to *close* input patterns
 481 presented within the same time window of $\Theta(t_m)$ rounds will be mapped to the same outputs.
 482 Recall that t_m is the memory duration which is a parameter of the network. A key component
 483 in our network is the *memory module* that remembers the association between each previously
 484 introduced pattern and its selected output for $\Theta(t_m)$ rounds. Roughly speaking, our network
 485 has two intermediate layers: an *association* layer and a *memory* layer (see Figure 3), which
 486 we describe below.

487 We first describe the construction by considering the case where a new pattern \bar{Z} is
 488 introduced (and no close pattern to it was introduced before). When \bar{Z} is presented to
 489 the network for the first time, it activates the association layer which contains r neurons
 490 $a_{i,1}, \dots, a_{i,r}$ for each output q_i . Let $\text{supp}(\bar{Z})$ be the non-zero entries of \bar{Z} . Since⁴ $|\text{supp}(\bar{Z})| \leq$
 491 ℓ it can activate at most $\ell \cdot k$ many neurons $a_{i,j}$ for every $j \in \text{supp}(\bar{Z})$ and $i \in \{1, \dots, k\}$.
 492 Every output q_i is connected to its association neurons $a_{i,1}, \dots, a_{i,r}$ and fires only if *many* of
 493 them fire.

494 Our construction will make sure that the number of active *association* neurons of a *taken*
 495 output (i.e., output already mapped to other pattern, far from \bar{Z}) will be small, which will
 496 prevent the firing of these outputs when a far pattern is presented. This will be provided
 497 due to the *memory module* appended to each output q_i which remembers the pattern (in
 498 fact the cluster of patterns) that were mapped to q_i in the *past*. For each j in the support
 499 of the pattern associated with q_i , the memory module corresponding to q_i and z_j inhibits
 500 all other association neurons associated with q_i , while activating the association a_{ij} . This
 501 association will be remembered – by the memory module – for at least $c_1 \cdot t_m$ rounds and at
 502 most $c_2 \cdot t_m$ rounds, for $c_1 < c_2$ with high probability.

503 For every available output q_i , all its association neurons $a_{i,j}$ for $j \in \text{supp}(\bar{Z})$ will start
 504 firing once \bar{Z} is presented, which will in turn activate q_i . To select exactly one output neuron
 505 among all the available ones, the output layer is connected via a lateral inhibition, where
 506 every neuron q_i inhibits all q_j for $j \geq i + 1$.

507 Overall, our sequential mapping module satisfies:

508 ► **Theorem 14** (The Sequential Mapping Module). *There exists a sequential mapping module*
 509 *with r input neurons, $\tilde{\Theta}(r \cdot k)$ auxiliary neurons, and k output neurons that for every pattern*
 510 *\bar{Z} that is introduced in round t satisfy the following with probability $1 - \delta$:*

- 511 (1) *The pattern \bar{Z} is mapped to one of the outputs q_1, \dots, q_k in round $t + 6$.*
 512 (2) *Any pair of close patterns \bar{Z}, \bar{Z}' introduced within a span of $c_1 \cdot t_m$ rounds are mapped*
 513 *to the same output neuron.*

⁴ As the WTA module picks at most one winning entry in each of the ℓ buckets.

514 (3) Any pair of far patterns \bar{Z}, \bar{Z}' introduced within a span of $c_1 \cdot t_m$ rounds are mapped to
515 different output neurons.

516 In addition, if a pattern \bar{Z} (or a pattern close to it) is not introduced for t_m rounds, then its
517 unique mapped output q_j is released after $c \cdot t_m$ rounds, for some constant $c \geq 1$.

518 5.1 Complete Network Description of the Sequential Mapping

519 Next we precisely describe the neurons and edge weights of the sequential mapping sub-
520 network.

521
522 **The association layer.** For each neuron z_i in the input layer, and each neuron q_j in
523 the output layer, we introduce an *association neuron* denoted as $a_{j,i}$. The neuron $a_{j,i}$ has
524 positive and negative incoming edges from the memory modules that is described in the next
525 paragraph. It also has a *positive* incoming edge from the neuron z_i with weight $w(z_i, a_{j,i}) = 2\ell$,
526 and bias $\beta(a_{j,i}) = (19/10)\ell - 1$. We set the connections to this neuron in a way that guarantees
527 it fires only if z_i fired in the previous round, and no other (far) pattern is already mapped to q_j .

528
529 **The memory modules.** For each neuron z_i in the input layer and output neuron q_j
530 we introduce a *memory of association* module $M_{j,i}$ which remembers the association of
531 neuron z_i and q_j . The memory module $M_{j,i}$ contains $c \cdot \log(\frac{1}{\delta'})$ excitatory neurons denoted
532 as $M_{j,i}^+$ where $\delta' = \delta/\ell$ and c is chosen to be a sufficiently large constant. For every $m \in M_{j,i}^+$
533 we introduce a *feedback* neuron f_m which starts exciting m once the memory module is being
534 activated. In addition, we introduce a delay chain $C_j^M = C_5(q_j)$ that starts at the output q_j
535 and ends at each of the neurons $m \in M_{j,i}^+$. Finally, the memory module contains two head
536 neurons, an excitatory neuron $m_{j,i}^+$ and an inhibitory neuron $m_{j,i}^-$.

Each excitatory neuron $m \in M_{j,i}^+$ has positive incoming edges from $a_{j,i}, q_j, C_j^M$, as well
as from the corresponding feedback neuron f_m with the following weights and bias

$$w(a_{j,i}, m) = 2\lambda, \quad \forall u \in C_j^M \quad w(u, m) = 2, \quad w(f_m, m) = \lambda \cdot (\chi + 2) + 9 \quad \beta(m) = 9 + 2 \cdot \lambda,$$

537 where $\chi = \log(t_m - 1)$. Note that if the feedback neuron f_m fired in the previous round, the
538 memory neuron m fires with probability at least $\frac{1}{1+e^{-\chi}} = 1 - 1/t_m$. The feedback neuron
539 f_m for $m \in M_{j,i}^+$ has *positive* incoming edges from m and $m_{j,i}^+$ with weights $w(m, f_m) =$
540 2 , $w(m_{j,i}^+, f_m) = 2$, and bias $\beta(f_m) = 3$. Hence, w.h.p. f_m fires iff m and $m_{j,i}^+$ fired in
541 the previous round. The excitatory head neuron $m_{j,i}^+$ has *positive* incoming edges from all
542 $m \in M_{j,i}^+$ with weights $w(m, m_{j,i}^+) = 2$ and bias $\beta(m_{j,i}^+) = c \cdot \log(1/\delta') + 1$. The head neuron
543 $m_{j,i}^-$ is an inhibitory copy of $m_{j,i}^+$ with the same incoming edges, bias and potential function.

544 Each association neuron $a_{j,i}$ has a *positive* incoming edge from the head memory neuron
545 $m_{j,i}^+$ with weight $w(m_{j,i}^+, a_{j,i}) = \ell$. In addition, $a_{j,i}$ has *negative* incoming edges from the in-
546 hibitory memory neurons $m_{j,i'}^-$ for every $i' = \{1, 2, \dots, k\} \setminus \{i\}$ with weights $w(m_{j,i'}^-, a_{j,i}) = -1$.
547 Note that w.h.p. $a_{j,i}$ fires in round t only if $z_i = 1$ in round $t - 1$. In case where there
548 are at least $1/10\ell$ memory modules $m_{j,i'}$ that inhibit $a_{j,i}$, it fires only if its own memory
549 module, namely, $M_{j,i}$ is active. To prevent a situation of partial memory where only part of
550 the memory modules associated with a pattern are released, if at most 0.9ℓ of the memory
551 modules $M_{j,1}, \dots, M_{j,r}$ are active, we activate the inhibition of these firing modules. For
552 that purpose, for every output q_j , we introduce 3 *deletion* neurons d_j^1, d_j^2, d_j^3 . The neurons
553 d_j^1, d_j^2 detect this situation and the inhibitor d_j^3 kills the partial memory. The deletion neuron
554 d_j^1 has incoming edges from all head neurons $m_{j,i}^+$ for $i = 1 \dots r$ with weights $w(m_{j,i}^+, d_j^1) = 2$
555 and bias $\beta(d_j^1) = 1$. Hence, w.h.p. d_j^1 fires in round t iff at least one memory module fired
556 in round $t - 1$. The second deletion neuron has incoming edges from all the inhibitor head

557 neurons $m_{j,i}^-$ for $i = 1 \dots r$ with weights $w(m_{j,i}^-, d_j^2) = -1$ and bias $\beta(d_j^2) = -0.9\ell + 1$. Thus,
 558 w.h.p. d_j^2 fires in round t iff at most 0.9ℓ memory module fired in round $t - 1$. The third
 559 deletion neuron d_j^3 has incoming edges from d_j^1 and d_j^2 with weights $w(d_j^1, d_j^3) = w(d_j^2, d_j^3) = 2$
 560 and bias $\beta(d_j^3) = 3$. Hence, d_j^3 fires in round t iff d_j^1 and d_j^2 fired in round $t - 1$. In
 561 addition, the head neurons $m_{j,i}^+, m_{j,i}^-$ have a *negative* incoming edge from d_j^3 with weight
 562 $w(d_j^3, m_{j,i}^+) = w(d_j^3, m_{j,i}^-) = -2c \log(1/\delta')$.

563
 564 **History neurons.** If an input pattern \bar{Z} is already mapped to an output neuron, our
 565 goal is to map every pattern close to \bar{Z} to the same output. To make sure that close patterns
 566 \bar{Z}, \bar{Z}' are indeed mapped to the same output, for each output neuron q_j we introduce an
 567 inhibitory *history* neuron h_j . The role of the history neuron is to take care of a situation
 568 where a pattern \bar{Z} is mapped to output q_j , but when a close pattern \bar{Z}' is presented later on,
 569 an output q_i for $i < j$ is free. Recall that in our construction, each pattern is mapped to
 570 the first available output. To do that, the network parameters of the history neurons are
 571 defined as follows. Each history neuron h_j has *positive* incoming edges from all associated
 572 excitatory memory neurons $m_{j,i}^+$ for $i = 1 \dots r$ with weights $w(m_{j,i}^+, h_j) = 1$. In addition, it
 573 has a *positive* incoming edge from the output neuron q_j with weight $w(q_j, h_j) = \ell$ and bias
 574 $\beta(h_j) = -(3/2)\ell - 1$. Thus, the history neuron h_j fires if the output neuron q_j fired and
 575 at least a large fraction of the memory modules corresponding to q_j are active (the latter
 576 indicates that q_j is indeed taken). The history neuron h_j then inhibits all the preceding
 577 output neurons q_1, \dots, q_{j-1} , preventing the input pattern from being mapped to a different
 578 output.

579
 580 **The output layer.** The output layer Q consists of excitatory neurons. In order to map
 581 the input pattern sequentially, for each $q_j \in Q$ we introduce an inhibitor output neuron q_j^-
 582 which inhibits the output neurons $q_{j'}$ for $j' \in \{j + 1, \dots, k\}$. The neuron q_j is connected to
 583 q_j^- via a delay chain of length 3 denoted as $C_j^I = C_3(q_j)$. The neuron q_j^- has incoming edges
 584 from C_j^I with weights 2, and a negative bias of $\beta(q_j^-) = 5$. Hence, w.h.p. q_j^- fires iff q_j fired
 585 for 3 consecutive rounds.

Each output neuron q_j has *positive* incoming edges from the association neuron $a_{j,i}$ for
 every $i = \{1, 2, \dots, k\}$. In addition, q_j has *negative* incoming edges from all preceding neurons
 q_i^- for $i < j$ and all successive history neurons h_i where $i > j$. The weights and bias are
 given by

$$w(a_{j,i}, q_j) = 2 \quad \forall i \in [r], \quad w(q_i^-, q_j) = -3\ell \quad \forall i < j, \quad w(h_i, q_j) = -3\ell \quad \forall i > j, \quad \beta(q_j) = \ell - 1$$

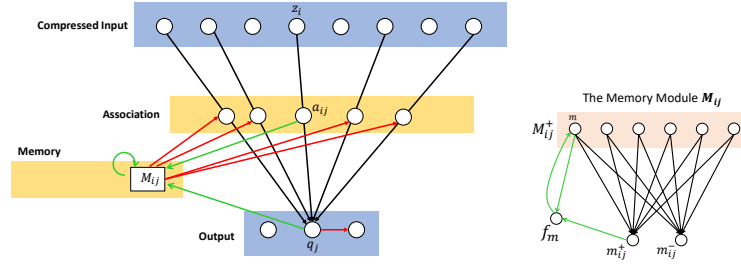
586 Note that q_j fires in round t only if at least $(1/2)\ell$ association neurons fired in round $t - 1$,
 587 and no history or inhibitor output neuron inhibit it.

588 As in previous sections, we assume that before the first round no neuron fires (i.e $v^0 = 0$
 589 for every neuron v in the network). Figure 4 illustrates the structure of the network and
 590 Figure 5 demonstrates the network flow with two inputs.

591 5.2 Network Dynamics

592 Before providing the detailed analysis of the network, we give a more detailed description
 593 of the network behavior in the two orthogonal cases: mapping close patterns to the same
 594 output and mapping far patterns to distinct outputs.

595
 596 **Introduction of a New Pattern \bar{X}_j .** A pattern \bar{X}_j is introduced in round t where
 597 q_1, \dots, q_{j-1} are already allocated. We will describe how \bar{X}_j is mapped to q_j . First, in Step



■ **Figure 4** Left: an illustration of the network. The green edges correspond to edges with positive weight where the red edges correspond to negative weights. For simplicity we omitted the history and deletion neurons as well as the rest of the association and memory modules. Right: The memory module and the feedback loop mechanism.

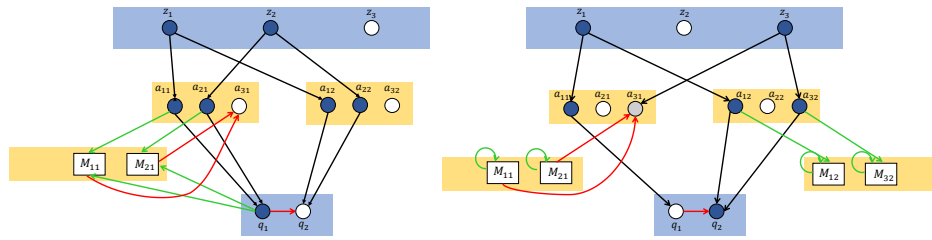
598 (1), \bar{X}_j is mapped to a vector $\bar{Y}_j = R_1(\bar{X}_j)$. In Step (2), \bar{Y}_j is mapped to a *binary* vector
 599 \bar{Z}_j which is the input to the sequential mapping sub-network. Let t' be the time in which
 600 \bar{Z}_j fires. This will cause the firing of the association layer in the following manner. Let
 601 $\bar{X}_1, \dots, \bar{X}_{j-1}$ be the patterns mapped to q_1, \dots, q_{j-1} .

- 602 ■ For every allocated neuron q_i , $i \leq j - 1$, and every entry $i_1 \in \text{supp}(Z_j) \setminus \text{supp}(Z_i)$, their
 603 association neuron a_{i,i_1} is inhibited by the memory modules M_{i,i_2} for every $i_2 \in \text{supp}(Z_i)$.
- 604 ■ Thus, for every allocated neuron q_i , when introducing Z_j , at most $|\text{supp}(Z_i) \cap \text{supp}(Z_j)| \leq$
 605 $0.1 \cdot \ell$ association neurons $a_{i,j'}$ are active.
- 606 ■ Since an output q_i fires only if at least $1/2\ell$ association neurons are active, q_i would not
 607 fire.
- 608 ■ For every free output q_i for $i \in \{j, \dots, k\}$, all the association neurons a_{i,i_1} for every
 609 $i_1 \in \text{supp}(Z_j)$ are now active. Hence, in the next round, all q_j, \dots, q_k fire.
- 610 ■ Since we have a lateral inhibition, q_j inhibits⁵ all other q_i for $i \in \{j + 1, \dots, k\}$.
- 611 ■ Only at the point where q_{j+1}, \dots, q_k are inhibited, the memory modules M_{j,i_1} of the
 612 winner output q_j start being active, for every $i_1 \in \text{supp}(Z_j)$. This memory module
 613 continues firing from that point on for $\Theta(t_m)$ rounds, even when X_j is not introduced.
- 614 ■ Each activated module M_{j,i_1} for every $i_1 \in \text{supp}(Z_j)$ inhibits each of the other association
 615 neurons a_{j,i_2} for every $i_2 \neq i_1$. In addition, each M_{j,i_1} excites its own association neuron
 616 a_{j,i_1} for $i_1 \in \text{supp}(Z_j)$, thus canceling the inhibition from the other M_{j,i_2} modules. As a
 617 result, the only inhibited association neurons are a_{j,i_2} for $i_2 \notin \text{supp}(Z_j)$.

618 **Re-Introduction of a Close-Pattern \bar{X}_j .** We now consider the situation where \bar{X}_j is
 619 introduced in round t , and a close-pattern $\bar{X}_{j'}$ was introduced in the past (e.g., in a window
 620 of $\Theta(t_m)$ rounds). We would like to show that \bar{X}_j will be mapped to the exact same output
 621 neuron $q_{j'}$ as $\bar{X}_{j'}$.

- 622 ■ For every allocated neuron q_i and every entry $i_1 \in \text{supp}(Z_j) \setminus \text{supp}(Z_i)$, their association
 623 neuron a_{i,i_1} is inhibited by the memory modules M_{i,i_2} for every $i_2 \in \text{supp}(Z_i)$.
- 624 ■ Thus, for every allocated neuron q_i for $i \neq j'$, when introducing Z_j , at most $|\text{supp}(Z_i) \cap$
 625 $\text{supp}(Z_j)| \leq 0.1 \cdot \ell$ association neurons a_{i,i_1} are active. As a result, q_{i_1} will not fire.
- 626 ■ In contrast, for the desired output neuron $q_{j'}$, only $|\text{supp}(Z_j) \setminus \text{supp}(Z_{j'})|$ association
 627 neurons are inhibited, while the remaining ones, namely, a_{j',i_1} for $i_1 \in \text{supp}(Z_j) \cap \text{supp}(Z_{j'})$
 628 are active. Since $|\text{supp}(Z_j) \cap \text{supp}(Z_{j'})|$ is sufficiently large, $q_{j'}$ will fire.

⁵ In fact, its inhibitor copy will do this inhibition.



■ **Figure 5** Left: network’s state where first pattern $(1, 1, 0)$ is presented. Since all outputs are free at that point, the pattern is mapped to the first output q_1 , which activates all its memory modules. Right: network description when the second input $(1, 0, 1)$ is presented. Because the memory modules $M_{1,1}$ and $M_{1,2}$ are active, the association neuron $a_{1,3}$ is inhibited and this q_1 will not fire. As a result, $(1, 0, 1)$ is mapped to q_2 , activating corresponding memory modules $M_{2,1}$ and $M_{2,3}$.

- 629 ■ Due to lateral inhibition of $q_{j'}$, all other free outputs $q_{i'}$ for $i' \geq j' + 1$ will not fire.
- 630 ■ It remains to show that all other *free* outputs q_i for $i \leq j' - 1$ will not be active. Recall
- 631 that these outputs have a lateral inhibition on $q_{j'}$ that starts inhibiting $q_{j'}$ within a
- 632 small number of rounds since the activation of q_i . It is therefore important to neutralize
- 633 these outputs before their inhibition on $q_{j'}$ comes into play. Indeed this is the reason for
- 634 introducing the delay to the lateral inhibition mechanism.
- 635 ■ To indicate the fact that $q_{j'}$ was already allocated to a pattern close to X_j , we have a
- 636 history neuron $h_{j'}$ that works as follows. It gets input from all the memory modules of
- 637 $q_{j'}$, as well as from $q_{j'}$ itself. Since the close patterns X_j and $X_{j'}$ have many entries in
- 638 common, sufficiently many memory modules of $q_{j'}$ will activate $h_{j'}$. For a free output
- 639 q_i for $i \leq j' - 1$, the memory modules of q_i are not active and hence the history neuron
- 640 would not be active.
- 641 ■ The history neuron $h_{j'}$ then inhibits all prior outputs q_i for $i \leq j' - 1$ just before their
- 642 lateral inhibition chain affects $q_{j'}$. In addition, the inhibition on q_i also occurs before the
- 643 memory modules of q_i start being active. That is, since we want to remember only the
- 644 association to the correct output $q_{j'}$, we delay the activation of the memory model. The
- 645 latter starts only after $q_{j'}$ fires for a consecutive constant number of rounds.

646 5.2.1 Correctness

647 The following definitions are useful in our context.

648 ► **Definition 15.** A pattern \bar{Z} is mapped to an output neuron q_j in round t if when presenting

649 \bar{Z} to the sequential mapping network in round $t - 1$, q_j is the only firing output neuron in

650 round t .

651 ► **Definition 16.** $M_{j,i}$ is active in round t , if its head neurons $m_{j,i}^+$, $m_{j,i}^-$ fired in round t .

652 In order to prove the main Theorem 14, we start by establishing useful auxiliary claims

653 and observations.

654 ▷ **Observation 17.** For every output neuron q_j if the number of active memory modules M_{ij}

655 in round t is between 1 and 0.9ℓ , then w.h.p. there are no active memory modules in round

656 $t + 3$.

657 **Proof.** For output neuron q_j if the number of active memory modules M_{ij} in round t is at
 658 least 1 w.h.p. the deletion neuron d_j^1 fires in round $t + 1$. If the number of active memory
 659 modules M_{ij} is also less than 0.9ℓ then w.h.p. d_j^2 fires in round $t + 1$ and therefore d_j^3 fires in
 660 round $t + 2$, inhibiting all memory modules M_{ij} for $i = 1, \dots, r$. ◀

661 \triangleright **Observation 18.** Given that the deletion neurons of output q_j did not fire in round $t - 1$,
 662 w.h.p. a memory module $M_{j,i}$ is active in round t iff at least $(c/2) \log(1/\delta')$ neurons $m \in M_{j,i}^+$
 663 fired in round $t - 1$.

Proof. Recall that a memory module $M_{j,i}$ is *active* in round t if the excitatory neuron $m_{j,i}^+$
 fired. The potential function of $m_{j,i}^+$ is given by

$$\text{pot}(m_{j,i}^+, t) = \sum_{m \in M_{j,i}^+} 2 \cdot m^{t-1} - 2c \log(1/\delta') (d_j^3)^{t-1} - c \log(1/\delta') + 1.$$

664 If at least $\frac{c}{2} \log(1/\delta')$ neurons in $M_{j,i}^+$ fired in round $t - 1$, the potential of $m_{j,i}^+$ in round t is at
 665 least 1 and the probability that $m_{j,i}^+$ fire in round t is at least $\frac{1}{1+e^{-1/\lambda}} \geq 1 - \Theta\left(\frac{\delta}{n \cdot k \cdot \Delta \cdot t_m \cdot \log 1/\delta}\right)$.
 666 On the other hand, if less than $\frac{c}{2} \log(1/\delta')$ neurons in $M_{j,i}^+$ fired, the potential of $m_{j,i}^+$ is at most
 667 -1 and the probability that $m_{j,i}^+$ fired in round t is at most $\frac{1}{1+e^{1/\lambda}} \leq \Theta\left(\frac{\delta}{n \cdot k \cdot \Delta \cdot t_m \cdot \log 1/\delta}\right)$. ◀

668 \triangleright **Claim 19.** If \bar{Z}_1, \bar{Z}_2 are close and \bar{Z}_2, \bar{Z}_3 are close, then \bar{Z}_1, \bar{Z}_3 are close.

669 **Proof.** Let $\bar{X}_1, \bar{X}_2, \bar{X}_3$ be the corresponding input patterns, where $\bar{Z}_i = R_2(\bar{X}_i)$ for $i \in$
 670 $\{1, 2, 3\}$. By the definition of the clustering instance, every pair of patterns \bar{X}_i, \bar{X}_j are either
 671 with relative distance at least $\Delta/2$ (i.e., if these patterns belong to different clusters), or have
 672 relative distance at most Δ/α (i.e., if they belong to the same cluster) for $\alpha = \Omega(\log(1/\Delta))$.

673 By Lemma 12, input patterns \bar{X}_i, \bar{X}_j that belong to different (resp., same) clusters are
 674 mapped to far (resp., close) vectors \bar{Z}_i, \bar{Z}_j . We therefore have that \bar{X}_1, \bar{X}_2 are in the same
 675 cluster, and also \bar{X}_2, \bar{X}_3 are in the same cluster, concluding that $\bar{X}_1, \bar{X}_2, \bar{X}_3$ are all in the
 676 same cluster. ◀

677 \triangleright **Claim 20.** For every $j \in [k]$ and $i \in [\ell]$ w.h.p. the memory module $M_{j,i}$ is active in round
 678 t given that it was not active in round $t - 3$, only if C_j^M and $a_{j,i}$ fired in round $t - 2$.

679 **Proof.** By Observation 18 $M_{j,i}$ is activated in round t only if at least $(c/2) \log(1/\delta')$ neurons
 680 $m \in M_{j,i}^+$ fire in round $t - 1$. Since $M_{j,i}$ was not active in round $t - 3$ all feedback neurons
 681 f_m for $m \in M_{j,i}^+$ was not active in round $t - 2$ and the potential of each $m \in M_{j,i}^+$ in round
 682 $t - 1$ is $\sum_{u \in C_j^M} 2 \cdot (u)^{t-2} + 2\lambda \cdot (a_{ij})^{t-2} - 9 - 2\lambda$. Hence, if C_j^M and $a_{j,i}$ fired in round $t - 2$,
 683 in the next round the potential of each $m \in M_{j,i}^+$ is at least 1 and m fires in round $t - 1$
 684 with probability at least $1 - \Theta\left(\frac{\delta}{n \cdot k \cdot \Delta \cdot t_m \cdot \log 1/\delta}\right)$. Thus, by Chernoff bound w.h.p. at least
 685 $(c/2) \log(1/\delta')$ neurons $m \in M_{j,i}^+$ fired in round $t - 1$.

686 On the other hand if C_j^M and $a_{j,i}$ did not fire together in round $t - 2$, the potential of
 687 every $m \in M_{j,i}^+$ in round $t - 1$ is at most -2λ and m fires with probability at most $\frac{1}{1+e^2}$.
 688 Using Chernoff bound and choosing c to be sufficiently large, we conclude that $(c/2) \log(1/\delta')$
 689 neurons $m \in M_{j,i}^+$ fire in round $t - 1$ with probability at most δ' . ◀

690 \triangleright **Observation 21.** For every output q_j at each round w.h.p. the number of memory modules
 691 $M_{j,i}$ that are active is at most ℓ .

692 **Proof.** Since every pattern has at most ℓ non zero entries, in each round at most ℓ association
 693 neurons $a_{j,i}$ fire. By Claim 20, at each round at most ℓ memory modules $M_{j,i}$ are activated

694 for the first time. If in round $t - 3$ more than 0.1ℓ memory modules were active, the only
 695 association neurons firing in round $t - 2$ correspond to the activated memory modules and
 696 therefore w.h.p. no new modules are activated in round t . Else, by Observation 17 w.h.p.
 697 the deletion neuron d_j^3 kills the active memory modules and no memory module is active in
 698 round t . ◀

699 Using the same arguments, since the deletion neurons erase the partial memory, we can also
 700 conclude that for every output neuron all its active memory modules correspond to the same
 701 input pattern.

702 ▷ **Observation 22.** For each output neuron q_i in each round if it has active memory modules,
 703 there exists an input pattern \bar{Z} s.t if $M_{i,j}$ is active then $j \in \text{supp}(\bar{Z})$.

704 ▷ **Claim 23.** If \bar{Z} is mapped to q_j in round t , with probability greater than $1 - \delta$ at least 0.8ℓ
 705 memory modules $M_{j,i}$ where $i \in \text{supp}(\bar{Z})$ are active for $c_1 \cdot t_m$ consecutive round starting
 706 from round $t + 8$.

707 **Proof.** Let \bar{Z} be a pattern mapped to q_j in round t . Recall that we assume persistence and
 708 therefore w.h.p. \bar{Z} is also mapped to q_j in rounds $t + 1$ to $t + 8$.

709 ■ First we argue that at least 0.8ℓ of the association neurons $a_{j,i}$ for $i \in \text{supp}(\bar{Z})$ fire in
 710 round $t + 6$. From Observation 17 either there where no memory modules corresponding
 711 to q_j active before \bar{Z} was introduced or at least 0.9ℓ ⁶. If there where no memory modules
 712 active, all association neurons $a_{j,i}$ for $i \in \text{supp}(\bar{Z})$ fire starting round $t + 1$ ahead as long
 713 as \bar{Z} persist. Otherwise, since q_j fired in round $t + 7$, we conclude that at least 0.5ℓ
 714 association neurons $a_{j,i}$ fired in round $t + 6$. The association neurons that fired are
 715 from the support of \bar{Z} and together with Observation 22 we conclude that the pattern
 716 previously mapped to q_j is close to \bar{Z} and at least 0.8ℓ association neurons fired in rounds
 717 $t + 6$ (due to Lemma 12).

■ For $i \in \text{supp}(\bar{Z})$ for which a_{ij} fired in rounds $t + 6$, we now calculate the probability that
 M_{ij} is active in round $t + 8$. By Observation 18 its enough to calculate the probability
 that at least $(c/2) \log(1/\delta')$ neurons $m \in M_{j,i}^+$ fired in round $t + 7$. The potential function
 of every $m \in M_{j,i}^+$ is given by

$$\text{pot}(m, t) = \sum_{u \in C_j^M} 2 \cdot (u)^{t-1} + 2\lambda \cdot (a_{ij})^{t-1} + (9 + \lambda \cdot (2 + \chi)) \cdot (f_m)^{t-1} - 9 - 2\lambda.$$

718 Since q_j fires in rounds t to $t + 7$, the delay chain C_j^M fired in round $t + 6$, and the
 719 probability m fires in round $t + 7$ is at least $1 - \Theta(\frac{\delta}{n \cdot k \cdot \Delta \cdot t_m \cdot \log 1/\delta})$. Using Chernoff bound
 720 with probability greater than $1 - \delta/3\ell$ at least $\frac{c \log(1/\delta')}{2}$ neurons in $M_{j,i}$ fire in round
 721 $t + 7$ and the head memory neuron $m_{j,i}^+$ fires in round $t + 8$.

722 ■ Next we calculate the probability that $m \in M_{j,i}^+$ fires $c_1 t_m$ consecutive rounds starting
 723 round $t + 8$ given that $m_{j,i}^+$ fires in round $t + 8$. Since $m_{j,i}^+$ fired in round $t + 8$, for every
 724 $m \in M_{j,i}^+$ that fired in round $t + 8$, the feedback neuron f_m is activated in round $t + 9$ and m
 725 fires in round $t + 10$ with probability at least $1 - 1/t_m$. Hence, the probability $m \in M_{j,i}^+$ fires
 726 in rounds $t + 8, t + 9$ and $c_1 t_m$ consecutive rounds is at least $(1 - \Theta(\frac{\delta}{n \cdot k \cdot \Delta \cdot t_m \cdot \log 1/\delta}))^2 \cdot (\frac{1}{e^{c_1}})$.
 727 We chose c_1 such that this is greater than $1/2$. Thus, using Chernoff bound and a large
 728 enough c (depending on c_1) the probability that at least $\frac{c \log(1/\delta')}{2}$ neurons $m \in M_{j,i}^+$ fire
 729 in rounds $t + 8, t + 9$ and then for $c_1 t_m$ consecutive rounds is at least $1 - \delta'/3 = 1 - \delta/3\ell$.

⁶ up too ± 3 rounds, but since we assume persistence its ok.

- Summing things up, the probability $M_{j,i}$ is active for $c_1 \cdot t_m$ consecutive rounds from round $t + 8$ ahead is at least the probability that $m_{j,i}^+$ fired in round $t + 8$ and $\frac{c \log(1/\delta')}{2}$ neurons in $M_{j,i}^+$ fires $c_1 t_m$ consecutive rounds starting round $t + 8$. By union bound this probability is greater than

$$1 - 3 \cdot (\delta/3\ell) = 1 - \delta/\ell.$$

Thus, we conclude that the probability all 0.8ℓ modules $M_{j,i}$ s.t $a_{j,i}$ fired in round $t + 6$ are active for $c_1 \cdot t_m$ consecutive rounds is greater than $1 - \delta$.

732

733 We are now ready to prove the correctness of the sequential mapping step.

734 Proof of Theorem 14

735 **Proof.** We start by proving the 3 main properties of the network. Given a pattern \bar{Z}
736 introduced in round t we will show:

- 737 (1) \bar{Z} is mapped to one of the outputs q_1, \dots, q_k in round $t + 6$
- 738 (2) For any pattern \bar{Z}' which is *close* to \bar{Z} and was introduced within a span of $c_1 \cdot t_m$
739 rounds from t , \bar{Z} and \bar{Z}' are mapped to the same output neuron.
- 740 (3) For any pattern \bar{Z}' which is *far* from \bar{Z} and was introduced within a span of $c_1 \cdot t_m$
741 rounds from t , \bar{Z} and \bar{Z}' are mapped to a different output neuron.

By induction on the order of arrival of the patterns. Let \bar{Z} be the first pattern arrived in round 0. We show that \bar{Z} is mapped to the first (available) neuron q_1 in round 6. For every $i \in \text{sup}(\bar{Z})$ the potential function of the association neuron $a_{1,i}$ is given by:

$$\text{pot}(a_{1,i}, t) = 2\ell(z_i)^{t-1} + \ell(m_{1,i}^+)^{t-1} - \sum_{j \neq i} (m_{1,j}^-)^{t-1} - (19/10)\ell - 1.$$

742 Since \bar{Z} is the first pattern seen, no neuron has fired in round zero and $\text{pot}(a_{1,i}, 1) =$
743 $(1/10)\ell - 1 > 1$, and w.h.p. each $a_{1,i}$ for $i \in \text{sup}(\bar{Z})$ fires in round 1.

Since q_1 is the first output, no preceding output neuron inhibits it, and its potential is:

$$\text{pot}(q_1, t) = \sum_{i=1}^r (2 \cdot a_{1,i})^{t-1} - \sum_{i=2}^k 3\ell h_i - \ell + 1.$$

744 By Claim 13, w.h.p. each input pattern \bar{Z} (to the sequential mapping network) has at
745 least 0.98ℓ non-zero entries (and at most ℓ). Therefore, at least 0.98ℓ association neurons
746 $a_{1,i}$ excite q_1 in round 2. Recall that the history neurons h_i fire only if at least $1/2$ of the
747 corresponding memory modules are active in the previous round. Hence w.h.p. in round 1,
748 no history neuron fires.

749 We conclude that q_1 fires in round 2 w.h.p. By Claim 20 every memory module $M_{i,j}$
750 becomes active only after having q_i firing for 5 consecutive rounds (due to the delay chain
751 C_i^M). For that reason, no memory module fires before round 5. Since the memory neurons
752 are not active, $a_{1,i}$ keeps firing in rounds 1 to 6, and q_1 keeps firing in rounds 2 to 7. Since
753 q_1 is connected to q_1^- via a delay chain C_1^I of length 3, starting round 5 (and as long as q_1
754 fires), the inhibitor q_1^- inhibits all other output neuron q_i for $i \geq 2$. Thus, for every $i \geq 2$ the
755 potential of q_i in round 6 is at most $\ell - 2\ell - 1/2\ell + 1 < -1$. As a result, for $i \geq 2$ neuron q_i
756 does not fire starting round 6.

757 We next argue that at this point, no memory modules are yet active and consequently the
758 history neurons are inactive as well. This is due to the fact that the delay in the inhibition

759 of q_i by q_1 is *shorter* than the delay chain C_i^M that starts at q_i and ends at the memory
 760 modules. Thus q_i is inhibited before its memory modules are activated. We conclude that if
 761 \bar{Z} is observed, starting from round 6, the output neuron q_1 is the only active output neuron,
 762 and \bar{Z} is *mapped* to q_1 .

763 Assume the claim holds for the first $i - 1$ presented patterns, we next consider the i^{th}
 764 pattern \bar{Z} presented in round t .

765 ■ We first show that for every \bar{Z}' that is far from \bar{Z} introduced in round $t' \in [t - c_1 \cdot t_m, t - 1]$,
 766 the pattern \bar{Z} will be mapped to a different output. By the induction assumption, \bar{Z}'
 767 is mapped in round $t' + 6$ to some output neuron, q_j . Since \bar{Z}' was introduced within
 768 $c_1 \cdot t_m$ rounds, by Claim 23 at least 0.8ℓ many memory modules $M_{j,i}$ for $i \in \text{sup}(\bar{Z}')$ are
 769 active it round t .

770 Let \bar{X} , \bar{X}' be the inputs corresponding to \bar{Z} and \bar{Z}' respectively. From Lemma 12,
 771 $\|\text{sup}(\bar{Z}) \cap \text{sup}(\bar{Z}')\| \leq 0.1\ell$. By Observation 21, the number of active memory modules
 772 $M_{j,i}$ in round t is at most ℓ . Thus, the number of active memory modules $M_{j,i}$ in round
 773 t for $i \in \text{sup}(\bar{Z})$ is at most $0.2\ell + 0.1\ell = 0.3\ell$.

774 For every association neuron $a_{j,i}$ whose memory module $M_{j,i}$ is inactive in round t , there
 775 are at least 0.8ℓ memory modules that inhibit it. Therefore its potential is $2\ell - 8/10\ell -$
 776 $(19/10)\ell + 1 < -1$, and w.h.p. it does not fire in round $t + 1$. Overall, at most 0.3ℓ
 777 association neurons $a_{j,i}$ start firing from round $t + 1$ and as long as the pattern persists.
 778 We conclude that q_j will stop firing from round $t + 2$.

779 ■ Next we show that two close patterns \bar{Z}' and \bar{Z} , introduced within a span of $c_1 \cdot t_m$
 780 rounds are mapped to the same output. First note that by Claim 19, all patterns that
 781 are close to \bar{Z} are close to each other. Hence, by the induction assumption, all patterns
 782 close to \bar{Z} introduced within the last $c_1 t_m$ rounds were mapped to the same output.
 783 We now consider a pattern \bar{Z}' close to \bar{Z} introduced in round $t' \in [t - c_1 \cdot t_m, t - 10]$.
 784 By the induction assumption, the pattern \bar{Z}' was mapped to output q_j in round $t' + 6$.
 785 By claim 23, 0.8ℓ many memory modules $M_{j,i}$ are active in round $t' + 7 < t$ onward
 786 (i.e., for $\Theta(t_m)$ rounds). Combining with Lemma 12 because \bar{Z} is close to \bar{Z}' , at least
 787 0.7ℓ many memory modules $M_{j,i}$, $i \in \text{sup}(\bar{Z})$ are active in round t . Since there are at
 788 most ℓ many active memory modules associated with q_j in round t , the potential of the
 789 association neuron $a_{j,i}$ for which the memory neuron is active in round $t + 1$ is at least
 790 $2\ell + \ell - (\ell - 1) - 1.9\ell - 1 = 0.1\ell > 1$. We have that at least 0.7ℓ many association neuron
 791 $a_{j,i}$ fire in round $t + 1$, leading to the firing of q_j in round $t + 2$.

792 Next, because at least 0.8ℓ memory modules $M_{j,i}$ are active from round $t + 3$ ahead, the
 793 history neuron h_j fires, and by that inhibits all output neurons q_i for $i \leq j - 1$ starting
 794 from round $t + 4$. Recall that every inhibitor neuron q_i^- for $i \leq j - 1$ starts firing only
 795 after the delay chain C_i^I fired, i.e. after 3 rounds that q_i fired. Hence the history neuron
 796 h_j inhibits every q_i for $i \leq j - 1$, just before q_i^- starts firing. We next show that no
 797 other q_i fires for $i \geq j + 1$. This holds since q_j^- inhibits any such q_i in round $t + 6$ via
 798 the delay chain C_j^I . Finally, we show that q_j continues firing as long the pattern persists.
 799 Because at least 0.5ℓ of the association neurons $a_{j,i}$ are firing, and no preceding inhibitor
 800 q_i^- is currently firing (thanks to the history neuron h_j), it remains to show that no other
 801 history neuron h_i for $i \neq j$ inhibits q_j . By the induction assumption, all patterns close
 802 to \bar{Z} were mapped to q_j . Hence if for some other output q_i for $i \neq j$, at least 0.5ℓ of its
 803 associated memory modules are active, by Observation 17 at least 0.9ℓ memory modules
 804 are active. Since the pattern \bar{Z}'' that was mapped to q_i is far from \bar{Z} , we have that at
 805 most 0.2ℓ many memory modules $M_{i,i'}$ for $i' \in \text{sup}(\bar{Z})$ are active, thus q_i does not fire
 806 and consequently h_i does not fire.

807 ■ We now consider the case of a newly presented pattern, i.e., no close pattern to it has
 808 been presented in the last $\Theta(t_m)$ rounds. We will show that in such a case, \bar{Z} will be
 809 mapped to the left-most available output q_j , where by available we mean that no memory
 810 module $M_{j,i}$ is active in round t . Let $q_{i_1}, q_{i_2} \dots q_{i_s}$ be the available output neurons in
 811 round t . Hence all the association neurons $a_{i_1,i}$ for $i \in \text{sup}(\bar{Z})$ start firing in round $t + 1$.
 812 This is because no memory module $M_{i_1,j}$ is active. Thus, in round $t + 2$ the output
 813 neuron q_{i_1} starts firing. As for the unavailable neurons q_j , by Observation 17 at least
 814 0.9ℓ memory modules $M_{j,i}$ are active and by Observation 22 they are associated with a
 815 pattern \bar{Z}' which was mapped to q_j . The pattern \bar{Z}' is far from \bar{Z} (\bar{Z} is a new pattern)
 816 and therefore at most 0.2ℓ memory module $M_{j,i}$ for $i \in \text{sup}(\bar{Z})$ are active in round t .
 817 Hence, at most 0.2ℓ association neurons $a_{j,i}$ fire starting round $t + 1$ and w.h.p. q_j will
 818 not fire starting round $t + 2$ (and also no history neuron inhibits q_{i_1}). Since we assume
 819 persistence, and due to the delay in the activation of the memory modules, q_{i_1} fires also
 820 in rounds $t + 3$ and $t + 4$, and in round $t + 5$ the inhibitor $q_{i_1}^-$ starts firing, inhibiting all
 821 the successive output neurons q_j for $j \geq i_1 + 1$.

822 In order to finish the proof of Theorem 14 we will prove the following Lemma.

823 ► **Lemma 24** (Reset, Clearance of Memory). *Let \bar{Z} be a pattern last introduced in round t and*
 824 *mapped to q_j . If no close pattern \bar{Z}' is introduced in rounds $[t, t + c_2 \cdot t_m]$, then q_j is released*
 825 *in some round $\tau \leq t + c_2 t_m$, i.e., all memory modules $M_{j,i}$ stop firing with probability greater*
 826 *than $1 - \delta$.*

827 **Proof.** Let \bar{Z} be a pattern last introduced in round t and mapped to q_j . By Observation 17
 828 if in some round τ there are less than 0.9ℓ memory modules corresponding to q_j firing, after
 829 3 round 0 memory modules are active and w.h.p. q_j is released. As long as there are at least
 830 0.9ℓ memory modules firing, since all patterns introduced in rounds t to $t + c_2 t_m$ are far
 831 from \bar{Z} by the same arguments used in Lemma 14 starting from round $t + 1$ less than 0.5ℓ
 832 association neurons associated with q_j fire and q_j will not fire for $c_2 t_m$ consecutive rounds
 833 starting from round $t + 2$ (as long as it is not already released). Thus, from round $t + 7$
 834 ahead w.h.p. all neurons in the delay chain C_j^M do not fire.

835 Therefore, the probability neuron $m \in M_{j,i}^+$ fires in round $\tau \in [t + 8, t + c_2 t_m]$ given that
 836 f_m did not fire in round $\tau - 1$ is at most $\Theta(\frac{\delta}{\log \frac{1}{\delta \cdot n \cdot k \cdot \Delta \cdot t_m}})$. Moreover, by union bound the
 837 probability that there exists a neuron $m \in M_{j,i}^+$ that fired in some round $\tau \in [t + 8, t + c_2 t_m]$
 838 given that f_m did not fire in round $\tau - 1$ is at most $\delta/2\ell$.

839 Next we calculate the probability at least half of the neurons $m \in M_{j,i}^+$ fire for $c_2 t_m$
 840 consecutive rounds. Because the delay chain C_j^M do not fire starting round $t + 7$ the potential of
 841 each neuron $m \in M_{j,i}^+$ in round $t' \in [t + 7, t + c_2 t_m]$ is bounded by $\lambda \cdot (\chi + 2) = \lambda(\log(t_m - 1) + 2)$.
 842 Hence, the probability $m \in M_{j,i}^+$ fires in round t' is at most $1 - \frac{1}{e^{2(t_m - 1) + 1}} < 1 - \frac{1}{e^{2t_m}}$. We
 843 conclude that the probability a neuron $m \in M_{j,i}^+$ fires for $c_2 t_m$ consecutive rounds is at most
 844 e^{c_2/e^2} which for $c_2 > e^2 \log(3)$ is less than $1/3$. Using Chernoff bound and a sufficient large c
 845 (constant depending on c_2) the probability that at least $(c/2) \log(1/\delta')$ neurons in $M_{j,i}^+$ fire
 846 for $c_2 \cdot t_m$ consecutive rounds starting round $t + 7$ is at most $\delta/2\ell$.

847 If $m_{j,i}^+$ fires for $c_2 t_m$ consecutive rounds starting round $t + 8$, by Observation 18 at each
 848 round at least $1/2$ of the neurons in $M_{j,i}^+$ fired. Given that no neuron $m \in M_{j,i}^+$ fires in
 849 round $\tau \in [t + 8, t + c_2 t_m]$ unless f_m fired in round $\tau - 1$, the head neuron $m_{j,i}^+$ fires for
 850 $c_2 t_m$ consecutive rounds only if at least $1/2$ of the neurons in $M_{j,i}^+$ fires for $c_2 t_m$ consecutive
 851 rounds. Thus we conclude that $m_{j,i}^+$ fired for $c_2 t_m$ consecutive rounds starting round $t + 8$
 852 with probability at most $\delta/(2\ell) + \delta/(2\ell) = \delta/\ell$. Note that by Observation 21 at most ℓ

853 memory neurons M_{ij} are active at each round and using union bound we conclude that with
 854 probability at least $1 - \delta$ the output neuron q_j is release in round $\tau < t + c_2 t_m$ ◀

855 This concludes Theorem 14 and therefore also Theorem 3. ◀

856 — References —

- 857 1 Jayadev Acharya, Arnab Bhattacharyya, and Pritish Kamath. Improved bounds for universal
 858 one-bit compressive sensing. In *2017 IEEE International Symposium on Information Theory*
 859 (*ISIT*), pages 2353–2357, 2017.
- 860 2 Zeyuan Allen-Zhu, Rati Gelashvili, Silvio Micali, and Nir Shavit. Sparse sign-consistent
 861 Johnson–Lindenstrauss matrices: Compression with neuroscience-based constraints. *PNAS*,
 862 2014.
- 863 3 Cornelia I Bargmann and Eve Marder. From the connectome to brain function. *Nature*
 864 *Methods*, 10(6):483–490, 2013.
- 865 4 Ella Bingham and Heikki Mannila. Random projection in dimensionality reduction: applica-
 866 tions to image and text data. In *Proceedings of the seventh ACM SIGKDD international*
 867 *conference on Knowledge discovery and data mining*, pages 245–250. ACM, 2001.
- 868 5 Burton H Bloom. Space/time trade-offs in hash coding with allowable errors. *Communications*
 869 *of the ACM*, 13(7):422–426, 1970.
- 870 6 Petros T Boufounos and Richard G Baraniuk. 1-bit compressive sensing. In *42nd Annual*
 871 *Conference on Information Sciences and Systems (CISS 2008)*., pages 16–21. IEEE, 2008.
- 872 7 Christos Boutsidis, Anastasios Zouzias, and Petros Drineas. Random projections for k -means
 873 clustering. In *Advances in Neural Information Processing Systems 23 (NIPS)*, 2010.
- 874 8 Emmanuel J Candes and Terence Tao. Near-optimal signal recovery from random projections:
 875 Universal encoding strategies? *IEEE transactions on information theory*, 52(12):5406–5425,
 876 2006.
- 877 9 Sophie JC Caron, Vanessa Ruta, LF Abbott, and Richard Axel. Random convergence of
 878 olfactory inputs in the drosophila mushroom body. *Nature*, 497(7447):113, 2013.
- 879 10 Chi-Ning Chou, Kai-Min Chung, and Chi-Jen Lu. On the algorithmic power of spiking neural
 880 networks. *arXiv preprint arXiv:1803.10375*, 2018.
- 881 11 Kenneth L Clarkson and David P Woodruff. Low rank approximation and regression in input
 882 sparsity time. In *Proceedings of the 45th Annual ACM Symposium on Theory of Computing*
 883 (*STOC*), 2013.
- 884 12 Michael B Cohen, Sam Elder, Cameron Musco, Christopher Musco, and Madalina Persu.
 885 Dimensionality reduction for k -means clustering and low rank approximation. In *Proceedings*
 886 *of the 47th Annual ACM Symposium on Theory of Computing (STOC)*, 2015.
- 887 13 Graham Cormode and Shan Muthukrishnan. An improved data stream summary: the
 888 count-min sketch and its applications. *Journal of Algorithms*, 55(1):58–75, 2005.
- 889 14 Robert Coultrip, Richard Granger, and Gary Lynch. A cortical model of winner-take-all
 890 competition via lateral inhibition. *Neural Networks*, 5(1):47–54, 1992.
- 891 15 Anirban Dasgupta, Ravi Kumar, and Tamás Sarlós. Fast locality-sensitive hashing. In
 892 *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and*
 893 *data mining*, pages 1073–1081. ACM, 2011.
- 894 16 Sanjoy Dasgupta, Timothy C Sheehan, Charles F Stevens, and Saket Navlakha. A neural data
 895 structure for novelty detection. *Proceedings of the National Academy of Sciences*, 115(51):13093–
 896 13098, 2018.
- 897 17 Sanjoy Dasgupta, Charles F Stevens, and Saket Navlakha. A neural algorithm for a fundamental
 898 computing problem. *Science*, 358(6364):793–796, 2017.
- 899 18 Mayur Datar, Nicole Immorlica, Piotr Indyk, and Vahab S Mirrokni. Locality-sensitive hashing
 900 scheme based on p -stable distributions. In *Proceedings of the twentieth annual symposium on*
 901 *Computational geometry*, pages 253–262. ACM, 2004.

- 902 19 David L Donoho. Compressed sensing. *IEEE Transactions on information theory*, 52(4):1289–
903 1306, 2006.
- 904 20 Surya Ganguli and Haim Sompolinsky. Compressed sensing, sparsity, and dimensionality in
905 neuronal information processing and data analysis. *Annual Review of Neuroscience*, 2012.
- 906 21 Simon S Haykin. *Neural networks and learning machines*, volume 3. Pearson, 2009.
- 907 22 Yael Hitron and Merav Parter. Counting to ten with two fingers: Compressed counting with
908 spiking neurons. *ESA*, 2019. URL: <http://arxiv.org/abs/1902.10369>, arXiv:1902.10369.
- 909 23 John J Hopfield, David W Tank, et al. Computing with neural circuits- a model. *Science*,
910 233(4764):625–633, 1986.
- 911 24 Laurent Jacques, Jason N Laska, Petros T Boufounos, and Richard G Baraniuk. Robust 1-bit
912 compressive sensing via binary stable embeddings of sparse vectors. *IEEE Transactions on*
913 *Information Theory*, 59(4):2082–2102, 2013.
- 914 25 Robert T Knight. Contribution of human hippocampal region to novelty detection. *Nature*,
915 383(6597):256, 1996.
- 916 26 Christof Koch and Shimon Ullman. Shifts in selective visual attention: towards the underlying
917 neural circuitry. In *Matters of intelligence*, pages 115–141. Springer, 1987.
- 918 27 Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 2015.
- 919 28 Robert A. Legenstein, Wolfgang Maass, Christos H. Papadimitriou, and Santosh Srinivas
920 Vempala. Long term memory and the densest k-subgraph problem. In *9th Innovations in*
921 *Theoretical Computer Science Conference, ITCS 2018, January 11-14, 2018, Cambridge, MA,*
922 *USA*, pages 57:1–57:15, 2018.
- 923 29 Andrew C Lin, Alexei M Bygrave, Alix De Calignon, Tzumin Lee, and Gero Miesenböck.
924 Sparse, decorrelated odor coding in the mushroom body enhances learned odor discrimination.
925 *Nature neuroscience*, 17(4):559, 2014.
- 926 30 Adi Livnat and Christos Papadimitriou. Evolution and learning: used together, fused together.
927 a response to watson and szathmáry. *Trends in Ecology & Evolution*, 31(12):894–896, 2016.
- 928 31 Nikos K Logothetis. What we can do and what we cannot do with fMRI. *Nature*, 453(7197):869,
929 2008.
- 930 32 Nancy Lynch and Cameron Musco. A basic compositional model for spiking neural networks.
931 *arXiv preprint arXiv:1808.03884*, 2018.
- 932 33 Nancy Lynch, Cameron Musco, and Merav Parter. Computational tradeoffs in biological
933 neural networks: Self-stabilizing winner-take-all networks. In *Proceedings of the 8th Conference*
934 *on Innovations in Theoretical Computer Science (ITCS)*, 2017.
- 935 34 Nancy Lynch, Cameron Musco, and Merav Parter. Neuro-RAM unit with applications to
936 similarity testing and compression in spiking neural networks. In *Proceedings of the 31st*
937 *International Symposium on Distributed Computing (DISC)*, 2017.
- 938 35 Nancy Lynch, Cameron Musco, and Merav Parter. Spiking neural networks: An algorithmic
939 perspective. In *5th Workshop on Biological Distributed Algorithms (BDA 2017)*, July 2017.
- 940 36 Wolfgang Maass. Networks of spiking neurons: the third generation of neural network models.
941 *Neural Networks*, 10(9):1659–1671, 1997.
- 942 37 Wolfgang Maass. On the computational power of winner-take-all. *Neural computation*,
943 12(11):2519–2535, 2000.
- 944 38 Christos H Papadimitriou and Santosh S Vempala. Cortical learning via prediction. In
945 *Conference on Learning Theory*, pages 1402–1422, 2015.
- 946 39 Christos H Papadimitriou and Santosh S Vempala. Random projection in the brain and
947 computation with assemblies of neurons. In *10th Innovations in Theoretical Computer Science*
948 *Conference (ITCS 2019)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018.
- 949 40 Narender Ramnani and Adrian M Owen. Anterior prefrontal cortex: insights into function
950 from anatomy and neuroimaging. *Nature Reviews. Neuroscience*, 5(3):184, 2004.
- 951 41 Charan Ranganath and Gregor Rainer. Cognitive neuroscience: Neural mechanisms for
952 detecting and remembering novel events. *Nature Reviews Neuroscience*, 4(3):193, 2003.

- 953 42 Tamas Sarlos. Improved approximation algorithms for large matrices via random projections.
 954 In *Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science*
 955 (*FOCS*), 2006.
- 956 43 Olaf Sporns, Giulio Tononi, and Rolf Kötter. The human connectome: a structural description
 957 of the human brain. *PLoS Computational Biology*, 1(4):e42, 2005.
- 958 44 Leslie G Valiant. *Circuits of the Mind*. Oxford University Press on Demand, 2000.
- 959 45 Leslie G Valiant. Memorization and association on a realistic neural model. *Neural computation*,
 960 17(3):527–555, 2005.
- 961 46 Leslie G Valiant. Capacity of neural networks for lifelong learning of composable tasks. In
 962 *Proceedings of the 58th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*,
 963 pages 367–378, 2017.
- 964 47 Santosh S Vempala. *The random projection method*, volume 65. American Mathematical
 965 Society, 2005.

966 **A Additional Proofs: Random Projection**

967 We first prove Lemma 5, that a Chi-squared distribution is nearly uniform within a constant
 968 number of standard deviations from its mean.

969 ▷ **Lemma 5.** Let \mathcal{D}_p be the Chi-squared distribution with p degrees of freedom. For any
 970 c with $1 \leq c < p^{1/2}$ there are constants c_ℓ, c_u (depending on c) such that, for any interval
 971 $[r_1, r_2] \subseteq [p - cp^{1/2}, p + cp^{1/2}]$, we have:

$$972 \frac{c_\ell(r_2 - r_1)}{p^{1/2}} \leq \Pr_{x \sim \mathcal{D}_p} [x \in [r_1, r_2]] \leq \frac{c_u(r_2 - r_1)}{p^{1/2}} \quad 973$$

974 That is, \mathcal{D}_p is roughly uniform on the range $[p - cp^{1/2}, p + cp^{1/2}]$.

975 **Proof.** It is well known that \mathcal{D}_p has mean p , density $d(x) = \frac{1}{2^{p/2}\Gamma(p/2)} x^{p/2-1} e^{-x/2}$. Since we
 976 assume $p^{1/2} > c \geq 1$ we have $p \geq 2$ and the distribution has mode $p - 2$. Additionally, we
 977 have $p - cp^{1/2} > 0$. So for $x \in [p - cp^{1/2}, p + cp^{1/2}]$ we can bound:

$$978 d(x) \leq d(p - 2) = \frac{1}{2^{p/2}\Gamma(p/2)} (p - 2)^{p/2-1} e^{-p/2+1} \leq \frac{1}{\Gamma(p/2)} \cdot \left(\frac{p}{2e}\right)^{p/2-1} \quad 979$$

980 By Stirling's approximation, $\Gamma(p/2) \geq \sqrt{\frac{2\pi}{p/2}} \left(\frac{p}{2e}\right)^{p/2}$ which gives:

$$981 d(x) \leq \sqrt{\frac{p}{4\pi}} \cdot \frac{2e}{p} = \frac{e}{\sqrt{\pi} \cdot p^{1/2}}. \quad 982 \quad (2)$$

983 On the other side, since $p - cp^{1/2} > 0$, and since the density of the Chi-squared distribution
 984 is monotonically decreasing as x moves further from the mode $p - 2$ either left or right:

$$985 d(x) \geq \min(d(p - cp^{1/2}), d(p + cp^{1/2})). \quad 986 \quad (3)$$

987 We lower bound each term in the minimum.

$$988 d(p - cp^{1/2}) = \frac{1}{2^{p/2}\Gamma(p/2)} (p - cp^{1/2})^{p/2-1} e^{-p/2+(c/2)p^{1/2}} \\ 989 = \frac{1}{2\Gamma(p/2)} \left(\frac{p}{2e}\right)^{p/2-1} \cdot \left(1 - \frac{c}{p^{1/2}}\right)^{p/2-1} \cdot e^{(c/2)p^{1/2}-1} \quad 990$$

991 Again using Stirling's approximation, and a similar argument to the proof of (2), for some
 992 constant c_1 , $\frac{1}{2\Gamma(p/2)} \left(\frac{p}{2e}\right)^{p/2-1}$ is lower bounded by $\frac{c_1}{p^{1/2}}$. Thus,

$$\begin{aligned}
 993 \quad d(p - cp^{1/2}) &\geq \frac{c_1}{p^{1/2}} \cdot \left(1 - \frac{c}{p^{1/2}}\right)^{p/2-1} \cdot e^{(c/2)p^{1/2}-1} \\
 994 &\geq \frac{c_1}{p^{1/2}} \cdot \left(1 - \frac{c}{p^{1/2}}\right)^{\left(\frac{p^{1/2}}{c}-1\right) \cdot (c/2)p^{1/2}} \cdot e^{(c/2)p^{1/2}-1} \cdot \left(1 - \frac{c}{p^{1/2}}\right)^{(c/2)p^{1/2}-1} \\
 995 &\geq \frac{c_1}{p^{1/2}} \frac{1}{e^{(c/2)p^{1/2}}} \cdot e^{(c/2)p^{1/2}-1} \cdot \left(1 - \frac{c}{p^{1/2}}\right)^{(c/2)p^{1/2}-1} \\
 996 &\geq \frac{c_1}{ep^{1/2}} \cdot \left(1 - \frac{c}{p^{1/2}}\right)^{\left(\frac{p^{1/2}}{c}-1\right) \cdot (c^2/2)+(c^2/2-1)} \\
 997 &\geq \frac{c_1 \cdot e^{c^2/2}}{ep^{1/2}} \cdot \left(1 - \frac{c}{p^{1/2}}\right)^{c^2/2-1} \geq \frac{c'}{p^{1/2}} \tag{4}
 \end{aligned}$$

999 for some constant c' that depends on c . We give a similar bound for $p + cp^{1/2}$.

$$\begin{aligned}
 1000 \quad d(p + cp^{1/2}) &= \frac{1}{2\Gamma(p/2)} \left(\frac{p}{2e}\right)^{p/2-1} \cdot \left(1 + \frac{c}{p^{1/2}}\right)^{-p/2-1} \cdot e^{(c/2)p^{1/2}-1} \\
 1001 &\geq \frac{c_1}{p^{1/2}} \cdot \left(1 + \frac{c}{p^{1/2}}\right)^{-p/2-1} \cdot e^{(c/2)p^{1/2}-1} \\
 1002 &= \frac{c_1}{p^{1/2}} \cdot \left(1 + \frac{c}{p^{1/2}}\right)^{-\frac{p^{1/2}}{c} \cdot (c/2)p^{1/2}} \cdot e^{(c/2)p^{1/2}-1} \cdot \left(1 + \frac{c}{p^{1/2}}\right)^{-1} \\
 1003 &\geq \frac{c}{ep^{1/2}} \cdot \left(1 + \frac{c}{p^{1/2}}\right)^{-1} \geq \frac{c'}{p^{1/2}} \tag{5}
 \end{aligned}$$

for some c' . Combining (4) and (5) with (3) and (2) gives that there exist constants c_ℓ, c_u such that for all $x \in [p - cp^{1/2}, p + cp^{1/2}]$,

$$\frac{c_\ell}{p^{1/2}} \leq d(x) \leq \frac{c_u}{p^{1/2}}.$$

1005 Thus for any r_1, r_2 :

$$1006 \quad \frac{c_\ell(r_2 - r_1)}{p^{1/2}} \leq \Pr_{x \sim \mathcal{D}_p} [x \in [r_1, r_2]] \leq \frac{c_u(r_2 - r_1)}{p^{1/2}},$$

1008 completing the lemma. ◀

1009 We next give a complete proof of Lemma 7.

1010 A.1 Proof of Lemma 7

1011 Since each $[A_b \bar{X}](i)$ is a Chi-squared random variable with p degrees of freedom, which has
 1012 median $\leq p$, each $[A_b \bar{X}](i)$ is upper bounded by $p \leq \tau = p + 2p^{1/2}$ with probability $\geq 1/2$.
 1013 Thus, by Lemma 5 applied with $c = 2$, conditioned on $[A_b \bar{X}](i) \leq p + 2p^{1/2}$, there is some c_ℓ
 1014 with:

$$1015 \quad \Pr \left[[A_b \bar{X}](i) \in [p, p + 2p^{1/2}] \mid [A_b \bar{X}](i) \leq p + 2p^{1/2} \right] \geq \frac{c_\ell \cdot 2p^{1/2}}{p^{1/2}} = 2c_\ell.$$

1016

1017 Thus, for large enough constant c_1 and $m = c_1$, with probability at least $\frac{199}{200}$, we have
 1018 $i_{1,b}(\bar{X}) \neq 0$ and $[A_b \bar{X}](i_{1,b}(\bar{X})) \geq p$. Call this event \mathcal{E}_1 . Condition on the event that \mathcal{E}_1
 1019 occurs and, in particular, that $[A_b \bar{X}](i_{1,b}(\bar{X})) = x$ for any $x \in [p, p + 2p^{1/2}]$. Call this event
 1020 $\mathcal{E}_{1,x}$. Then for all $j \neq i_{1,b}(\bar{X})$, $[A_b \bar{X}](j)$ is an independent Chi-squared random variable
 1021 with p degrees of freedom conditioned on either 1) $[A_b \bar{X}](j) \leq x$ or 2) $[A_b \bar{X}](j) \geq p + 2p^{1/2}$.
 1022 Since $[A_b \bar{X}](j) \leq p \leq x$ with probability at least $1/2$, this conditioning at most doubles the
 1023 density at any one value. Thus, by Lemma 5,

$$1024 \quad \Pr \left[[A_b \bar{X}](j) \in \left[x - \frac{p^{1/2}}{c_2 m}, x \right] \mid \mathcal{E}_{1,x} \right] \leq \frac{2c_u \cdot \frac{p^{1/2}}{c_2 m}}{p^{1/2}}.$$

1026 By a union bound, we thus have:

$$1027 \quad \Pr \left[\exists j : [A_b \bar{X}](j) \in \left[x - \frac{p^{1/2}}{c_2 m}, x \right] \mid \mathcal{E}_{1,x} \right] \leq \frac{2c_u \cdot \frac{p^{1/2}}{c_2}}{p^{1/2}} = \frac{2c_u}{c_2}.$$

1029 Setting c_2 sufficiently large ensures that this quantity is bounded by $\frac{1}{200}$. Thus, by a union
 1030 bound with the probability that \mathcal{E}_1 occurs, with probability $\geq \frac{99}{100}$: $i_{1,b}(\bar{X}) \neq 0$ and 2) no
 1031 $[A_b \bar{X}](j)$ falls in $\left[x - \frac{p^{1/2}}{mc_2}, x \right] = \left[[A_b \bar{X}](i_{1,b}(\bar{X})) - \frac{p^{1/2}}{mc_2}, [A_b \bar{X}](i_{1,b}(\bar{X})) \right]$. This completes
 1032 the proof.

1033 A.2 Proof of Lemma 8

1034 We first use the relative distance assumption to give a basic claim:

1035 \triangleright **Claim 25.** Write \bar{X}_1, \bar{X}_2 as $\bar{X}_1 = \chi + \delta_1$ and $\bar{X}_2 = \chi + \delta_2$ where $\chi \in \{0, 1\}^n$ is the
 1036 common vector with $\chi(i) = 1$ iff $\bar{X}_1(i) = \bar{X}_2(i) = 1$. Note that since $\|\bar{X}_1\| = \|\bar{X}_2\| = p$ we
 1037 have $\|\delta_1\| = \|\delta_2\|$. Letting $\Delta = \mathcal{RD}(\bar{X}_1, \bar{X}_2)$,

$$1038 \quad \frac{\|\delta_1\|}{p} = \frac{\Delta}{2}.$$

1040 **Proof.** We can write:

$$1041 \quad \Delta = \mathcal{RD}(\bar{X}_1, \bar{X}_2) = \frac{\|\bar{X}_1 - \bar{X}_2\|}{p} = \frac{\|\delta_1 - \delta_2\|}{p} = \frac{\|\delta_1\| + \|\delta_2\|}{p}.$$

1043 The claim follows since $\|\delta_1\| = \|\delta_2\|$. \blacktriangleleft

1044 \triangleright **Claim 26.** For $i \in [2]$ and $j \in [m] \cup 0$ let \mathcal{E}_j be the event that $j = i_{1,b}(\bar{X}_1)$. With
 1045 probability $\geq 999/1000$ over the choice of $A_b \chi$, for all j we have:

$$1046 \quad \Pr[\mathcal{E}_j \mid A_b \chi] \leq \frac{1}{16}.$$

Proof. Let $\Delta = \mathcal{RD}(\bar{X}_1, \bar{X}_2)$ and assume for simplicity that $\Delta \leq 1$ (we will later see that
 it is easy to remove this assumption). By Claim 25, $\|\delta_1\| = \|\delta_2\| \leq \frac{p}{2}$ and thus $\|\chi\| \geq \frac{p}{2}$. For
 a constant c_3 (to be set later) sub-divide the range $[\|\chi\| - c_3 p^{1/2}, \|\chi\| + c_3 p^{1/2}]$ into $\frac{1}{\Delta^{1/2}}$
 subranges of width:

$$2c_3 p^{1/2} \Delta^{1/2} = 2\sqrt{2}c_3 \|\delta_1\|^{1/2},$$

1048 where the equality follows from Claim 25. By Lemma 5 (applied with the constant c
 1049 in the Lemma set to c_3) for any $i \in [m]$, $[A_b \chi](i)$ falls into each range with probability
 1050 $\Theta(c_3 \cdot \Delta^{1/2})$. Thus, by a standard Chernoff bound, for $m = \frac{c_1 \log 1/\Delta}{\sqrt{\Delta}}$ for sufficiently large

1051 c_1 , with probability 1999/2000 over the choice of A_b , at least c_4 indices of $A_b\chi$ fall within
 1052 each bucket where c_4 is a constant to be set later. Note that c_1 depends on c_3, c_4 . Call the
 1053 event that c_4 indices fall into each bucket $\mathcal{E}_{full-buckets}$. Additionally, as argued in Lemma
 1054 7, for sufficiently large m , the maximum value $[A_b\bar{X}_1](i_{1,b}(\bar{X}_1))$ below $p + p^{1/2}$ satisfies
 1055 $[A_b\bar{X}_1](i_{1,b}(\bar{X}_1)) \geq p$ with probability at least 1999/2000. Thus, with probability 1999/2000
 1056 over the choice of A_b ,

$$1057 \quad \Pr [[A_b\bar{X}_1](i_{1,b}(\bar{X}_1)) \geq p | A_b\chi] \geq 1999/2000. \quad (6)$$

Let \mathcal{E}_{good} be the event that both $\mathcal{E}_{full-buckets}$ and (6) hold. \mathcal{E}_{good} holds with probability
 $\geq 999/1000$ over the choice of A_b . First note that conditioning on \mathcal{E}_{good} , $\Pr[\mathcal{E}_0 | A_b\chi] \leq \frac{1}{2000}$,
 easily giving the claim for $j = 0$. We now consider $j \in [m]$. We consider any bucket,

$$R = \left\{ j : [A_b\chi](j) \in \left[r, r + 2\sqrt{2}c_3\|\delta_1\|^{1/2} \right] \right\},$$

1059 where r is be some integer multiple of $2\sqrt{2}c_3\|\delta_1\|^{1/2}$. Roughly, since each index in R has a
 1060 very similar value in $A_b\chi$, each has nearly the same likelihood of being the largest entry in
 1061 $A_b\bar{X}_1$ below $\tau = p + 2p^{1/2}$. Since $\mathcal{E}_{full-buckets}$ occurs, there are at least c_4 of these indices
 1062 and thus if c_4 is large, none has very high probability of being the largest entry. Formally,
 1063 we will show that, assuming \mathcal{E}_{good} holds, for each $j \in R$,

$$1064 \quad \Pr[\mathcal{E}_j | A_b\chi] \leq \frac{1}{16}. \quad (7)$$

1066 Since this bound holds for all buckets in the range $[\|\chi\| - c_3p^{1/2}, \|\chi\| + c_3p^{1/2}]$, it will give
 1067 the claim after arguing that no index with $A_b\chi$ falling outside this range is likely to have
 1068 $\mathcal{E}(1, j)$ occur either.

1069 **Indices in Buckets:** Each entry of $A_b\delta_1$ is identically distributed as an independent Chi-
 1070 squared random variable with $\|\delta_1\|$ degrees of freedom. Additionally, $A_b\delta_1$ is independent of
 1071 $A_b\chi$ since δ_1 and χ have disjoint supports. Consider $j \in R$ with $\Pr[[A_b\bar{X}_1](j) \geq \tau | A_b\chi] \geq$
 1072 $15/16$. In this case, since $\mathcal{E}(1, j)$ can only hold if $[A_b\bar{X}_1](j) \leq \tau$, (7) trivially holds.

1073 Next consider $j \in R$ with $\Pr[[A_b\bar{X}_1](j) \geq \tau | A_b\chi] \leq 15/16$. By Lemma 6 there is some c
 1074 with:

$$1075 \quad \Pr [[A_b\delta_1](j) \geq \|\delta_1\| + c\|\delta_1\|^{1/2}] = \frac{1}{64},$$

1077 or equivalently since $[A_b\bar{X}_1](j) = [A_b\chi](j) + [A_b\delta_1](j)$:

$$1078 \quad \Pr [[A_b\bar{X}_1](j) \geq [A_b\chi](j) + \|\delta_1\| + c\|\delta_1\|^{1/2} | A_b\chi] = \frac{1}{64},$$

1080 Setting $r_2 = \min([A_b\chi](j) + \|\delta_1\| + c\|\delta_1\|^{1/2}, \tau)$ we thus have that

$$1081 \quad \Pr [[A_b\bar{X}_1](j) \in [r_2, \tau] | A_b\chi] \leq \frac{1}{64}. \quad (8)$$

1083 Additionally, by Lemma 5 there is some r_1 with $r_2 - r_1 = \Theta(\|\delta_1\|^{1/2})$ such that:

$$1084 \quad \Pr [[A_b\bar{X}_1](j) \in [r_1, r_2] | A_b\chi] = \frac{1}{32}.$$

1086 Since for all $j' \in R$, $|[A_b\chi](j) - [A_b\chi](j')| \leq 2\sqrt{2}c_3\|\delta_1\|^{1/2} = O(\|\delta_1\|^{1/2})$ we have $r_2 =$
 1087 $[A_b\chi](j') + O(\|\delta_1\|^{1/2})$ and thus again by Lemma 5, for all $j' \in R$:

$$1088 \quad \Pr [[A_b\bar{X}_1](j') \in [r_1, r_2] | A_b\chi] = \Omega(1).$$

1090 If we set the constant c_4 large enough, since assuming \mathcal{E}_{good} , $|R| \geq c_4$ we have:

$$1091 \quad \Pr[\exists j' \in R \setminus j : [A_b \bar{X}_1](j') \in [r_1, r_2] | A_b \chi] \geq \frac{31}{32}.$$

1093 If this event holds, we can only have $\mathcal{E}(1, j)$ occur if $[A_b \bar{X}_1](j)$ falls in $[r_2, \tau]$, which by (8)
1094 occurs with probability $\leq \frac{1}{64}$ conditioned on $A_b \chi$. Thus by a union bound we have:

$$1095 \quad \Pr[\mathcal{E}_j \mid A_b \chi] \leq \frac{1}{16},$$

1097 giving (7) in this case.

1098 **Indices Outside Buckets:** We now consider indices not falling in any bucket: that is,
1099 j with $[A_b \chi](j) \leq \|\chi\| - c_3 p^{1/2}$ or $[A_b \chi](j) \geq \|\chi\| + c_3 p^{1/2}$. For the later, to have $\mathcal{E}_b(1, j)$
1100 occur we must have $[A_b \bar{X}_1](j) \leq \tau = p + 2p^{1/2}$ and thus $[A_b \delta_1](j) \leq \|\delta_1\| - (c_3 - 2)p^{1/2} \leq$
1101 $\|\delta_1\| - (c_3 - 2)\sqrt{2}\|\delta_1\|$. By Lemma 6, this occurs with probability $< 1/16$ for all $\|\delta_1\|$
1102 as long as we set c_3 large enough. Similarly, for j with $[A_b \chi](j) \leq \|\chi\| - c_3 p^{1/2}$, with
1103 probability $\geq 15/16$, we will have $[A_b \delta_1](j) \leq \|\delta_1\| + c_3 p^{1/2}$ and thus $[A_b \bar{X}_1](j) \leq p$. Since
1104 assuming \mathcal{E}_{good} , the maximum value of $A_b \bar{X}_1$ bounded by $\leq \tau$ is $\geq p$, if $[A_b \bar{X}_1](j) \leq p$,
1105 $\mathcal{E}_b(1, j)$ will not occur. Thus completes the argument in this case, giving that for all j with
1106 $[A_b \chi](j) \leq \|\chi\| - c_3 p^{1/2}$ or $[A_b \chi](j) \geq \|\chi\| + c_3 p^{1/2}$, $\Pr[\mathcal{E}_b(1, j) \mid A_b \chi] \leq \frac{1}{16}$.

1107 **Removing Bound on Δ :** Finally, we note that we can remove the assumption that $\Delta \leq 1$.
1108 If $\Delta \geq 1$ we can simply have χ encompass some of the non-shared entries in \bar{X}_1 until $\|\chi\| \geq \frac{p}{2}$
1109 and $\|\delta_1\| \leq \frac{p}{2}$ as desired. The bound will go through as argued up to constants, since we will
1110 still have $\frac{\|\delta_1\|}{p} = \Theta(\Delta)$ as in Claim 25 (note that we always have $\Delta \leq 2$).
1111 ◀

1112 We can now complete the proof of Lemma 8. We have:

$$1113 \quad \Pr[i_{1,b}(\bar{X}_1) = i_{1,b}(\bar{X}_2) \mid A_b \chi] = \sum_{j=0}^m \Pr[i_{1,b}(\bar{X}_1) = i_{1,b}(\bar{X}_2) = j \mid A_b \chi]$$

$$1114 \quad = \sum_{j=0}^m \Pr[i_{1,b}(\bar{X}_1) = j \mid A_b \chi] \cdot \Pr[i_{1,b}(\bar{X}_2) = j \mid A_b \chi]$$

$$1115 \quad (9)$$

1116 where the second line follows from the fact that $A_b \bar{X}_1$ and $A_b \bar{X}_2$ are independent conditioned
1117 on $A_b \chi$ since δ_1, δ_2 are disjoint vectors. By Claim 26, with probability $\geq 999/1000$ over the
1118 choice of $A_b \chi$ we can bound (9) by:

$$1119 \quad \Pr[i_{1,b}(\bar{X}_1) = i_{1,b}(\bar{X}_2) \mid A_b \chi] \leq \sum_{j=0}^m \Pr[i_{1,b}(\bar{X}_2) = j \mid A_b \chi] \cdot 1/16 = 1/16 \quad (10)$$

1121 where the last line follows simply since $\sum_{j=0}^m \Pr[i_{1,b}(\bar{X}_2) = j \mid A_b \chi] = 1$. Since (10) holds
1122 with probability $\geq 999/1000$ over the choice of $A_b \chi$, overall $\Pr[i_{1,b}(\bar{X}_1) = i_{1,b}(\bar{X}_2)] \leq$
1123 $1/16 + 1/1000$.

1124 Applying Claim 7 and a union bound gives that $i_{1,b}(\bar{X}_1) \neq i_{1,b}(\bar{X}_2)$ and the gaps between
1125 the largest and second largest entries of $A_b \bar{X}_1$ and $A_b \bar{X}_2$ (bounded by τ) are both at least
1126 $\geq \frac{p^{1/2}}{c_2 \cdot m}$ (or there is at most one such entry), and $[A_b \bar{X}_1](i_1(\bar{X}_1)), [A_b \bar{X}_2](i_1(\bar{X}_2)) \geq p$ with
1127 probability $\geq 1 - (1/16 + 1/1000) - 2/100 = .9165$, giving the lemma.

1128 **A.3 Proof of Lemma 10**

 1129 We now give the deferred proof of Lemma 10, which shows that two close inputs are likely to
 1130 have the same intermediate neuron with the maximum potential $\leq \tau$ in each bucket. We
 1131 restate the lemma below.

 1132 \triangleright **Lemma 10.** Let $\bar{X}_1, \bar{X}_2 \in \{0, 1\}^n$ be two vectors with $\mathcal{RD}(\bar{X}_1, \bar{X}_2) \leq \Delta/\alpha$. Consider our
 1133 construction with bucket size $m = \frac{c_1 \log(1/\Delta)}{\sqrt{\Delta}}$. Then for sufficiently large constants c_1, c_2 and
 1134 $\alpha = O(\log(1/\Delta)^4)$, for any $b \in [\ell]$, defining $i_{1,b}(\cdot)$ and $i_{2,b}(\cdot)$ as in Lemma 7, with probability
 1135 ≥ 0.97 :

- 1136
- \blacksquare
- $i_{1,b}(\bar{X}_1) = i_{1,b}(\bar{X}_2)$
- .
-
- 1137
- \blacksquare
- For both
- $j = 1, 2$
- :
- $i_{1,b}(\bar{X}_j) \neq 0$
- ,
- $[A_b \bar{X}_j](i_{1,b}(\bar{X}_j)) \geq p$
- , and
-
- 1138
- $[A_b \bar{X}_j](i_{1,b}(\bar{X}_j)) - [A_b \bar{X}_j](i_{2,b}(\bar{X}_j)) \geq \frac{p^{1/2}}{c_2 m}$
- or
- $i_{2,b}(\bar{X}_j) = 0$
- .

 1139 **Proof.** By Lemma 7, with probability $\geq 99/100$, for all $i \in [m] \setminus i_{1,b}(\bar{X}_1)$ with $[A_b \bar{X}_1](i) \leq \tau$:

1140
$$[A_b \bar{X}_1](i_{1,b}(\bar{X}_1)) - [A_b \bar{X}_1](i) = \Omega\left(\frac{p^{1/2}}{m}\right), \quad (11)$$
 1141

 1142 By a similar argument, with probability $\geq 99/100$, for all $i \in [m]$,

1143
$$|\tau - (A_b \bar{X}_1)_i| = \Omega\left(\frac{p^{1/2}}{m}\right). \quad (12)$$
 1144

Additionally, by standard sub-exponential concentration (as used in Lemma 6) with probability $\geq 99/100$, for both $j = 1, 2$ and all $i \in m$ we have $[A_b \delta_j](i) \in \|\delta_i\| \pm O(\log m \cdot \|\delta_i\|^{1/2})$. Note that $\log m = O(\log(1/\Delta))$. Additionally, by Claim 25, since $\mathcal{RD}(\bar{X}_1, \bar{X}_2) \leq \Delta/\alpha$ for $\alpha = O(\log(1/\Delta)^4)$, we have for both $i = 1, 2$, $\frac{\|\delta_i\|}{p} \leq \frac{\Delta}{2\alpha} = O\left(\frac{\Delta}{\log(1/\Delta)^4}\right)$. This gives that

$$O(\log m \cdot \|\delta_1\|^{1/2}) = O\left(\frac{\Delta^{1/2} p^{1/2}}{\log(1/\Delta)}\right) = O\left(\frac{p^{1/2}}{m}\right).$$

 1145 So for both $j = 1, 2$ and all $i \in m$, $(A_b \delta_j)_i \in \|\delta_i\| \pm O\left(\frac{p^{1/2}}{m}\right)$. So by (11) we have for all
 1146 $i \neq i_{1,b}(\bar{X}_1)$ with $[A_b \bar{X}_1](i) \leq \tau$:

1147
$$[A_b \bar{X}_2](i_{1,b}(\bar{X}_1)) - [A_b \bar{X}_1](i) =$$
 1148
$$[A_b \bar{X}_1](i_{1,b}(\bar{X}_1)) - [A_b \delta_1](i_{1,b}(\bar{X}_1)) + [A_b \delta_2](i_{1,b}(\bar{X}_1)) - [A_b \bar{X}_1](i) = \Omega\left(\frac{p^{1/2}}{m}\right).$$
 1149

1150 By (12) we also have,

1151
$$[A_b \bar{X}_2](i_{1,b}(\bar{X}_1)) = [A_b \bar{X}_1](i_{1,b}(\bar{X}_1)) - [A_b \delta_1](i_{1,b}(\bar{X}_1)) + [A_b \delta_2](i_{1,b}(\bar{X}_1)) \leq \tau.$$
 1152

 1153 and similarly, for all $i \neq i_{1,b}(\bar{X}_1)$ with $[A_b \bar{X}_1](i) \geq \tau$:

1154
$$[A_b \bar{X}_2](i) \geq \tau.$$
 1155

 1156 That is, $i_{1,b}(\bar{X}_1)$ is the largest entry of $A_b \bar{X}_2$ under τ , and thus $i_{1,b}(\bar{X}_1) = i_{1,b}(\bar{X}_2)$.

 1157 Applying Lemma 7 and a union bound gives the second claim with overall probability
 1158 $1 - 1/100 - 1/100 - 1/100 = 97/100$.

1159

◀

1160 **B Detailed Analysis of the Sparsification Step via WTA**

1161 **Proof of Claim 13**

1162 **Proof.** Let $i = \arg \max_{j: \bar{Y}(j) \leq \tau} \bar{Y}(j)$. For every neuron $j \in \{1, \dots, m\}$ in the input vector
 1163 \bar{Y} , let $\bar{R}(j)$ be the random variable that counts the number of rounds in which j fires in a
 1164 window of $T = \Theta(m^2 \log m)$ rounds. By the construction described above in which all j with
 1165 $\bar{Y}(j) \geq \tau$ are inhibited with very strong weight, $\bar{R}(j) = 0$ w.h.p. for all such j . Thus we focus
 1166 on j with $\bar{Y}(j) \leq \tau$. We show that if $\bar{Y}(i) - \bar{Y}(j) = \Omega\left(\frac{p^{1/2}}{m}\right)$ for $j \neq i$, then $\bar{R}(i) \gg \bar{R}(j)$
 1167 with probability at least $1 - \Theta(1/m)$.

1168 First, let \bar{P} be the vector of firing probabilities of each intermediate neuron induced by
 1169 the potentials in \bar{Y} (ignoring the entries that have been zero'd out since $\bar{Y}(j) \geq \tau$). By (1)
 1170 we have $\bar{P}(i) = \frac{1}{1+e^{-\bar{Y}(i)}}$. Letting $s(x) = 1/(1+e^{-x})$, we have $s'(x) \in [\frac{1}{2}, \frac{3}{4}]$ for $x \in [0, 1]$
 1171 and can see that if $\bar{Y}(i) - \bar{Y}(j) = \Omega(1/m)$, then also $s(\bar{Y}(i)) - s(\bar{Y}(j)) = \Omega(1/m)$. That is,
 1172 a gap of $\Omega(1/m)$ between $\bar{Y}(i)$ and $\bar{Y}(j)$ translates to a gap of $\Omega(1/m)$ between the firing
 1173 probabilities $\bar{P}(i)$ and $\bar{P}(j)$. To ensure that $\bar{Y}(i), \bar{Y}(j)$ are in $[0, 1]$ we can simply rescale the
 1174 weights of the random connection matrix A by $\frac{1}{2p}$ and shift them by p by adding a bias of p to
 1175 each intermediate neuron. By Corollary 9, before this shift and scaling, $\bar{Y}(i) \in [p, p + 2p^{1/2}]$,
 1176 so afterwards, $\bar{Y}(i) \in [0, 1]$. For all $j \neq i$, since by Corollary 9 we had $\bar{Y}(i) - \bar{Y}(j) = \Omega\left(\frac{p^{1/2}}{m}\right)$
 1177 we still have $\bar{Y}(i) - \bar{Y}(j) = \Omega\left(\frac{1}{m}\right)$ as required and thus $\bar{P}(i) - \bar{P}(j) = \Omega\left(\frac{1}{m}\right)$.

By Chernoff bound, with probability of at least $1 - c/m$,

$$\bar{R}(i) \geq T \cdot \bar{P}(i) - \sqrt{T \cdot \bar{P}(i) \cdot c \log m} \quad \text{and} \quad \bar{R}(j) \leq T \cdot \bar{P}(j) + \sqrt{T \cdot \bar{P}(j) \cdot c \log m}.$$

1178 Hence, with probability $1 - 2c/m$ we get that

$$\begin{aligned} 1179 \quad \bar{R}(i) - \bar{R}(j) &\geq T \cdot (\bar{P}(i) - \bar{P}(j)) - \sqrt{T \cdot \bar{P}(i) \cdot c \log m} - \sqrt{T \cdot \bar{P}(j) \cdot c \log m} \\ 1180 &\geq T \cdot (\bar{P}(i) - \bar{P}(j)) - 2\sqrt{T \cdot \bar{P}(i) \cdot \log m} = \Omega(T/m) - O(\sqrt{T \cdot \log m}) \\ 1181 &= \Omega(T/m), \end{aligned}$$

1182 by taking $T = c' \cdot m^2 \log m$ for a sufficiently large constant c' .

1183 Since the incoming weight of each neuron $y_{i,j}$ is $\bar{R}(i) - \bar{R}(j) = \omega(1)$, we get that $y_{i,j}$ fires
 1184 with probability of $1 - \Theta(1/m)$. By doing a union bound over all $m - 1$ neurons, and taking
 1185 large enough constants, we get that with probability at least $99/100$, all neurons $y_{i,j}$ fire for
 1186 every $j \neq i$. Hence, z_i is the only firing neuron in \bar{Z} . \blacktriangleleft

1187 Recall that in Step (1), every input vector \bar{X}_i is projected into ℓ vectors $\bar{Y}_{i,b} = A_b \cdot \bar{X}_i$
 1188 for every $b \in \{1, \dots, m\}$. On each such vector $\bar{Y}_{i,b}$ we apply the WTA circuit and get a
 1189 vector $\bar{Z}_{i,b}$. Let $\bar{Z}_i = \bar{Z}_{i,1} \circ \bar{Z}_{i,2} \circ \dots \circ \bar{Z}_{i,\ell}$ for $\ell = O(\log(t_m/\delta))$, where \circ denotes vector
 1190 concatenation.

1191 We conclude this section by showing that the relative gap between input patterns \bar{X}_i, \bar{X}_j
 1192 is reflected in their output vectors of the WTA circuit. By combining Claim 13 with Cor. 9
 1193 and 11, we prove Lemma 12 which completes the correctness of Step (II).

1194 **Proof of Lemma 12**

1195 First observe that for every input \bar{X}_i , there are at most ℓ non-zero entries in \bar{Z}_i since the
 1196 threshold gates fire only if there is a sufficient gap in the firing rates. (I) For a fixed pair \bar{X}_i, \bar{X}_j
 1197 of far patterns, let $B_{i,j}$ be the set of all buckets b where $\arg \max_{r \in [m]: \bar{Y}_{i,b}(r) \leq \tau} \bar{Y}_{i,b}(r) \neq$

1198 $\arg \max_{r \in [m]: \bar{Y}_{j,b}(r) \leq \tau} \bar{Y}_{j,b}(r)$ and the gap between largest and second largest entries in both
 1199 vectors $\bar{Y}_{i,b}$ and $\bar{Y}_{j,b}$ is $\Omega(p^{1/2}/m)$. By Cor. 9, with probability $1 - \delta$, for every pair of far
 1200 patterns \bar{X}_i, \bar{X}_j , $|B_{i,j}| \geq 0.9 \cdot \ell$.

1201 By Claim 13, if $\bar{Y}_{i,b}$ has a desired gap between the largest entry and other entries
 1202 then, with probability $p = 99/100$, $\bar{Z}_{i,b}$ has exactly one winning entry corresponding to
 1203 $\arg \max(\bar{Y}_{i,b})$. In expectation the vectors $\bar{Z}_{i,b}$ differ in $p \cdot |B_{i,j}|$ buckets. Thus by applying
 1204 Chernoff bound overall k^2 pairs, in 0.9ℓ of the buckets, the WTA picks a distinct winner for
 1205 the \bar{X}_i and \bar{X}_j patterns. Thus, $\text{supp}(\bar{Z}_i) \setminus \text{supp}(\bar{Z}_j) \geq 0.9\ell$.

1206 (II) For a fixed pair \bar{X}_i, \bar{X}_j of close patterns, let $B_{i,j}$ be the set of all buckets b where
 1207 $\arg \max_{r \in [m]: \bar{Y}_{i,b}(r) \leq \tau} \bar{Y}_{i,b}(r) = \arg \max_{r \in [m]: \bar{Y}_{j,b}(r) \leq \tau} \bar{Y}_{j,b}(r)$ and the gap between largest
 1208 and second largest entries in both vectors $\bar{Y}_{i,b}$ and $\bar{Y}_{j,b}$ is $\Omega(p^{1/2}/m)$. By Cor. 11 with
 1209 probability $1 - \delta$, for every pair of close patterns \bar{X}_i, \bar{X}_j , $|B_{i,j}| \geq 0.91 \cdot \ell$. By applying Claim
 1210 13 and Chernoff bound overall k^2 pairs, in at least $0.9 \cdot \ell$ of the buckets, the selected winner
 1211 is the same with probability of $1 - \delta$, implying that $\text{supp}(\bar{Z}_i) \cap \text{supp}(\bar{Z}_j) \geq 0.9 \cdot \ell$.