Available online at www.sciencedirect.com
ScienceDirect
Neural Networks 19 (2006) 1161–1167
Neural Networks
www.elsevier.com/locate/neunet
2006 Special Issue

Neural voting machines

Whitman Richards[a,b,*], H. Sebastian Seung[b,c], Galen Pickard[a]

[a] *Computer Science and Artificial Intelligence Lab (CSAIL), Massachusetts Institute of Technology, Bldg. 32-364, Cambridge, MA 02139, United States*
[b] *Brain and Cognitive Sciences Departments, Massachusetts Institute of Technology, Bldg. 46-5065, Cambridge, MA 02139, United States*
[c] *Howard Hughes Medical Institute, Massachusetts Institute of Technology, Bldg. 46-5065, Cambridge, MA 02139, United States*

Received 31 October 2005; accepted 8 June 2006

Abstract

A "Winner-take-all" network is a computational mechanism for picking an alternative with the largest excitatory input. This choice is far from optimal when there is uncertainty in the strength of the inputs, and when information is available about how alternatives may be related. For some time, the Social Choice community has recognized that many other procedures will yield more robust winners. The Borda Count and the pair-wise Condorcet tally are among the most favored. If biological systems strive to optimize information aggregation, then it is of interest to examine the complexity of networks that implement these procedures. We offer two biologically feasible implementations that are relatively simple modifications of classical recurrent networks.

Keywords: Information aggregation; Social networks; Decision-making; Winner-take-all; Borda Count; Condorcet voting; Domain knowledge; Neuroeconomics

1. Introduction

Information aggregation in neural networks is a form of collective decision-making. The winner-take-all procedure is a popular method of picking one of many choices among a landscape of alternatives (Amari & Arbib, 1977; Maass, 2000). In the social sciences, this is equivalent to choosing the plurality winner, which is but one of a host of procedures that could be used to choose winners from a set of alternatives. Saari (1994) points out many undesirable characteristics of this procedure. More importantly, in the presence of uncertainty about choices, the plurality winner is not the maximum likelihood choice, whereas either Borda or Condorcet are, depending upon whether or not the entire social order is desired (Young, 1995). To obtain a glimpse into some of the problems associated with winner-take-all, consider the analogy where the biological input is seen as a population of voters. Then the plurality winner – that outcome shared by most of the voters – only needs to receive more votes than any other alternative in the choice set. Hence it is possible for the winner to garner only a very small percentage of the total votes cast. In this case, uncertainty and errors in opinions can have a significant impact on outcomes, such as when only a few "on-the-fence" voters switch choices (Richards, 2005). We sketch two other procedures that yield more reliable and robust winners, and illustrate how they may be implemented in neural networks.

2. Plurality voting

To provide background, the winner-take-all procedure is recast as a simple voting machine. Let there be n alternative choices a_i with v_i of the voters preferring alternative a_i. The inputs to the n nodes in a neural network will then be the number of voters v_i sharing the same preference for a winner. The outcome is

$$\text{winner\_Plurality} = \text{argmax}_i\{v_i\} \tag{1}$$

which can be found using a winner-take-all network based on lateral inhibition. There are a number of implementations of the winner-take-all network; examples can be found in the references (Amari & Arbib, 1977; Xie, Hahnloser, & Seung, 2001). Note that no information about any correlated relationships among alternatives is captured in (1). In other

* Corresponding author at: Computer Science and Artificial Intelligence Lab (CSAIL), Massachusetts Institute of Technology, Bldg. 32-364, Cambridge, MA 02139, United States. Tel.: +1 617 253 5776; fax: +1 617 253 0415.
E-mail addresses: wrichards@mit.edu (W. Richards), seung@mit.edu (H.S. Seung), gpickard@mit.edu (G. Pickard).

0893-6080/$ - see front matter
doi:10.1016/j.neunet.2006.06.006

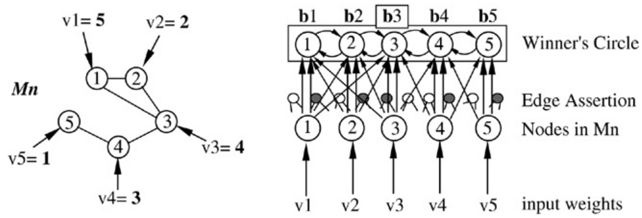

Fig. 1. A Borda* Count network for the domain model $\mathbf{M}_n$ at left. The input weights v_i are analogous to the number of voters who favor alternative a_i, which is represented as node i in the network. Each input node projects to all nodes in another layer called "the winner's circle". The direct projection to the node similarly labeled has weight two, as indicated by the bold double arrows. The input nodes also project with weight one to the remaining $n-1$ nodes in the winner's circle, but some of these are inhibited by interneurons in the intermediate edge assertion layer. The role of inhibition is to delete the weight-one projections to the winner's circle if there are no edges in the model $\mathbf{M}_n$. The nodes in the winner's circle then carry out a WTA computation on the selected inputs. (Not all recurrent connections are shown in the winner's circle.)

words, if there are correlations among the alternative choices a_i, this information will be ignored in the Plurality tally.

3. Borda method

To improve the informativeness (and robustness) of outcomes, we follow recommendations in Social Decision-Making, and relax the constraint that only first choices will be considered in the voting process. In Social Decision-Making, this is equivalent to including second (or higher) ranked preferences (Runkel, 1956; Saari, 1994; Saari & Haunsberger, 1991). The Borda Count (Borda, 1784) includes this information by assigning $n-1$ points to a voter's first choice, $n-2$ points to his second ranked alternative, and generally $n-k$ points to the kth ranked alternative. In the case of ties in a sequence of rankings, points that would otherwise be assigned to alternatives are averaged, and the average is given to each member in that set of ties. The maximum of these weighted sums is then taken as the winner. Note that implicit in this procedure is that the voter's ranking is an indication of the similarity or correlations among the alternatives, from the voter's viewpoint. Our main assumption is that the similarities or correlations among the n alternative choices are related by a domain model $\mathbf{M}_n$ that is held in common by all voters. For clarity, we take $\mathbf{M}_n$ to be an undirected graph, such as the one in Fig. 1. The preference rankings used by voters v_i are then dictated by the digraph $\mathbf{D}_i$ induced by $\mathbf{M}_n$, with the root of the digraph being the voter's first choice, namely a_i. In other words, each voter's ranking of alternatives is now not arbitrary, but is also reflecting information about choice relationships in the shared domain model $\mathbf{M}_n$ (Richards, 2001; Richards, McKay, & Richards, 1998, 2002). In our simple example, there are five voters favoring alternative a_1, with second choices of a_2 and a_3, and third choice a_4, etc. Similarly there is only one voter favoring a_5, with second choice a_4 and third choice a_3. For voter v_i, the level in the preference ordering for a_j is simply the smallest number of edges in the digraph $\mathbf{D}_i$ between a_i and a_j.

In effect, the role of $\mathbf{M}_n$ is to place conditional priors on relationships in the choice domain. Hence individual preference rankings will not be independent. This has consequences for how a Borda tally should be conducted. In the Appendix, we justify a tally that assigns $n-k$ points to each voter's kth choice, even when several alternatives have the same preference rank. This revision of the accepted Borda method is an important factor in insuring that prior information about relations among alternatives captured by $\mathbf{M}_n$ will be utilized to yield the best estimate of a true majority winner. We will denote our modification as the Borda* method, to highlight its difference from the classical Borda Count.

Although the shared model $\mathbf{M}_n$ has typically been represented as a graph, $\mathbf{G}_n$, it is more convenient to use the matrix $\mathbf{M}_{ij}$ where the entry "1" indicates the presence of the edge ij in $\mathbf{G}_n$ and 0 otherwise (Harary, 1969). For the graphical model of Fig. 1, we would have:

$$\mathbf{M}_{ij} = \begin{matrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{matrix}\,. \tag{2}$$

For simplicity, we assume that the edges of $\mathbf{M}_n$ are undirected, meaning that if alternative a_1 is similar to alternative a_2, then a_2 is equally similar to a_1. However, directed edges, or the inclusion of weighted edges, require only trivial modifications to the scheme. Again for simplicity, we will also restrict our preference orders to first, second and third choices. Because $\mathbf{M}_n$ is modeled here as a random graph, this restriction has little practical consequence for large random graphs; their diameter is almost always two (Harary, 1969), and hence any voter will need no more than three levels of preferences to rank all alternatives.

With $\mathbf{M}_n$ expressed as the matrix $\mathbf{M}_{ij}$ we can calculate the new Borda* tally that includes second choice opinions by defining a new voting weight v_i^* as

$$v_i^* = 2\, v_i + \sum_j \mathbf{M}_{ij}\, v_j \tag{3}$$

where now first-choice preferences are given twice the weight of the second-ranked choices, and third or higher ranked options have zero weight. The outcome is then

$$\text{winner\_Borda}^* = \text{argmax}_i\{v_i^*\}. \tag{4}$$

A neural network that executes this tally is shown at the right of Fig. 1. At the top (winner's circle) is a standard winner-take-all (WTA) layer. (The collection of WTA nodes in the "winner's circle" does not show all the recurrent connections.) Below this is an "edge assertion layer" that embodies the model $\mathbf{M}_{ij}$ and controls whether or not an input v_i will contribute to another node j such that $j \neq i$. Hence, in accord with the classical Borda count, the ith WTA node receives a synapse of strength 2 from the ith input node, which in turn is driven by v_i. This twice weighted input to a WTA node is depicted by heavy double arrows. The ith WTA node also receives synapses of strength $\mathbf{M}_{ij}$ from the jth input. These inputs are depicted by the slimmer arrows which are active only if there is an edge in $\mathbf{M}_n$.

This selection is carried out by the intermediate edge assertion layer. Here we need ${}_nC_2$ neurons, each representing whether or not there is an edge in $\mathbf{M}_n$ between nodes i and j, and accordingly inhibiting inputs from v_i to v_j in the winner's circle if edge $[v_i, v_j]$ is not in $\mathbf{M}_n$. This layer is the limiting factor in the complexity of the network. For the model illustrated, the highlighted WTA node 3 is the Borda* winner for the given inputs v_i. Note that the more common winner-take-all plurality procedure would pick node 1.

4. Rank vector

A disadvantage of *any* Borda Count is that a weighting on a voter's preferences is imposed depending on rank, or, equivalently, the minimum number of edge steps from a voter's first choice to another alternative. In our simple Borda* model using only first and second choice preferences, a weight of 2 was given to the first choice and 1 for the second. Let this bias be represented as the Borda* rank vector [2, 1, 0], or equivalently, as the normalized form [1, 0.5, 0], where the 0 is the weight applied to all preferences ranked after second choices. Then it is clear that the rank vector for the Plurality method is [1, 0, 0]. But we could also invent another vector [1, 1, 0] that would weight the "Top Two" choices equally. More generally, a normalized rank vector will have the form $[1, b, c]$ with $0 < b < 1$ and $c = 0$ for our simplified preference rankings. But now we see that the outcome of any Borda* procedure will depend on the choices for b, c. What justifies one weighting scheme over another?

To avoid specifying values for b, alternatives can be compared pairwise. Each voter would then simply pick the most preferred alternative of each pair — the one with the higher rank in his preference order. There are several procedures using pairwise comparisons, with the outcome depending upon whether or not there is one alternative that beats all others (Condorcet, 1785; Dodgson, 1876; Klamler, 2004; Ratliff, 2002). Here we follow Condorcet's proposal, where the winner is that alternative beating all others.

Definition. Let d_{ij} be the minimum number of edge steps between vertices i and j in $\mathbf{M}_n$, where each vertex corresponds to the alternatives a_i and a_j respectively.

Then a pairwise Condorcet score S_{ij} between alternatives a_i and a_j is given by

$$S_{ij} = \sum_k v_k \mathrm{Sign}[d_{jk} - d_{ik}] \qquad (5)$$

with the sign positive for the alternative a_i or a_j closer to a_k. Note that if a_i or a_j are equidistant from a_k, then Sign = 0 and the voting weight v_k does not contribute to S_{ij}.

Furthermore, as in the Borda* Count, we again impose a maximum on the value of d_{ij} of 2, which means that third or higher ranked alternatives do not enter into the tally.

A Condorcet winner is defined by

$$\text{winner\_Condorcet} = i \text{ iff } S_{ij} > 0 \text{ for all } j \text{ such that } j \neq i. \qquad (6)$$

Thus the Condorcet winner is a true majority outcome. But, whereas there is always a Borda* winner in the absence of ties, a unique Condorcet winner is not guaranteed. Failures occur when there is a social cycle among the top three or more alternatives.

Definition. A top cycle is present (and hence no unique winner) if there is some set of alternatives such that a_i beats a_j, a_j beats a_k, and a_k beats a_i, and every alternative not in the top cycle is beaten by at least one alternative in the top cycle.

Thus, although the Condorcet tally also utilizes information about relations among alternatives in the choice domain, it has two disadvantages over the Borda* method. First, there is no guaranteed winner (which may be appropriate if one seeks a clear consensus), and second it comes at a computational cost. For n alternatives, the Borda* computation is $O(n)$ if $\mathbf{G}_n$ is sparse and $O(n^2)$ if dense, whereas the Condorcet pair-wise calculation is $O(n^3)$. Superficially, then one might expect a neural network that calculates the Condorcet winner to be more complex than that for the Borda* winner.

5. Condorcet network

To reduce the computational complexity to $O(n^2)$, the trick is to choose a special subgraph of $\mathbf{G}_n$, namely $\mathbf{g}_k$, with $k \ll n$. Conceptually, the subgraph we choose is a ridge in the landscape of Borda* weights. The ridge consists of the k nodes in $\mathbf{G}_n$ with the highest Borda* scores (i.e. Eq. (3)). This choice is based on the observation that for connected random graphs of fixed edge probability with weights chosen from a uniform distribution, simulations show there is near 100% likelihood that the Condorcet winner will be among those alternatives with the top five Borda* scores. (The Appendix elaborates this observation.)

5.1. Specifics for the subgraph g_k

Let the Borda* rank vector be [2, 1, 0] as before, with the Borda* scores v_i^* for each vertex i in $\mathbf{G}_n$. Without loss of generality, label the vertices in $\mathbf{G}_n$ by the rank order of their Borda* score, with vertex $i = 1$ having the largest score. In cases where the Borda* scores are tied, simply choose the indexing arbitrarily among the tied vertices to create a total order.

Definition. $\mathbf{g}_k$ is the spanning subgraph of $\mathbf{G}_n$ containing the vertices with the top k Borda* scores.

Note that other definitions are possible. For example, we could require that $\mathbf{g}_k$ be a connected subgraph. In this case, for some $\mathbf{G}_n$, $\mathbf{g}_k$ may not include all the top k Borda* scores.

5.2. Sketch of a neural network

Here, our objective is to obtain a crude sense of the complexity of a plausible neural network that calculates a winner for $\mathbf{g}_k$. There are three design challenges: (1) finding $\mathbf{g}_k$, (2) computing the Condorcet winner for each pairwise comparison, and (3) to determine which alternative (node) beats all others. We assume that the maximum Borda ridge (nodes in $\mathbf{g}_k$) has already been found, perhaps using a scheme similar to that shown in Fig. 1. If $\mathbf{g}_k$ can be unconnected as defined, then a clipping algorithm might suffice to identify the k nodes in $\mathbf{G}_n$ with the highest Borda* scores. Alternatively, if we wish to impose a connectivity constraint on $\mathbf{g}_k$, then some form of a greedy algorithm beginning at vertex $i = 1$ would be more appropriate. Simulations based on random graphs $\mathbf{G}_n$, for $n = 40$ with edge probability 1/4 and weights chosen from a uniform distribution show that in over 96% of cases, the winners with $k = 8$ are the same regardless whether the definition of $\mathbf{g}_k$ is satisfied precisely, or found using a greedy algorithm. This equivalence might be expected, because the vertices with the highest Borda* scores will typically have the largest vertex degrees and hence the greatest connectivity. In either case, this step is of complexity $O(n^2)$.

The second challenge is to implement ${}_kC_2$ pairwise comparisons [Eq. (5)]. (As before, third or higher ranked alternatives will not enter into the tally.) The trick requires noting whether the vertices being compared are adjacent or not.

Consider first the case where two vertices in $\mathbf{g}_k$ are not adjacent in $\mathbf{G}_n$ (and hence also not adjacent in $\mathbf{g}_k$). Then we simply need to sum the weights of the neighbors to each vertex, plus the weight of the vertex itself, and then compare these two weight sums to determine the pairwise winner. Note that this is equivalent to using a different rank vector for each vertex, namely [1, 1, 0], and then picking that vertex with the largest score. If the two vertices being contested are adjacent, however, then note that the weight of each vertex will be added to the score of the competing vertex. Hence the weights of the vertices themselves will be cancelled if [1, 1, 0] rank vector is used. To correct for this cancellation, the weights of each vertex in the comparison must be doubled when these vertices are adjacent (Richards et al., 2002). This is the Borda* rank vector [2, 1, 0]. Hence, when calculating each pairwise Condorcet score, the rule is to use a Top-Two bias vector [1, 1, 0] when vertices are non-adjacent in $\mathbf{G}_n$, and to use the Borda* rank vector [2, 1, 0] when adjacent. This requires making explicit whether or not the ${}_kC_2$ edges in $\mathbf{g}_k$ are adjacent or not in $\mathbf{G}_n$.

The third challenge is to determine that node or vertex beating all others. As will be seen below, this can be handled easily by a logical AND of the WTA outputs from each pairwise comparison.

Fig. 2 depicts a $\mathbf{g}_k$ network with six layers. The computation is carried out as follows:

(i) the k-maximum Borda* ridge (nodes in $\mathbf{g}_k$) is given, as well as the neighborhood sums in $\mathbf{G}_n$ for each node in $\mathbf{g}_k$.

(ii) Activate the ${}_kC_2$ set of nodes in an "edge assertion" layer to make explicit which edges in $\mathbf{M}_n$ are present in $\mathbf{G}_n$ (as was done previously in the Borda* network). Note that this activation controls inputs to both the neighborhood sums layer, as well as to nodes in the comparator layer.

(iii) For every node in $\mathbf{g}_k$, find the sum of the weight of vertex i and its neighbors (neighborhood sums). Again, this step is analogous to that used in the Borda* network of Fig. 1, except at this stage the sum uses the "Top-Two" bias vector [1, 1, 0]. The second input for the weight of vertex i itself will be added in step (iv) depending upon edge connectivity in $\mathbf{G}_n$.

(iv) Project the activity of [1, 1, 0] neighborhood node sums onto each member of a pair of nodes in the "comparator" layer that has the same vertex label.

(v) Project the weight of vertex i itself onto the appropriate member of all pairs in the comparator layer, but only if the two vertices in the comparator layer are adjacent in $\mathbf{g}_k$ (as controlled by the edge assertion activations in (ii)).

(vi) Use a WTA procedure to select the winner of each pairwise comparison in the comparator layer, and send either a "0"

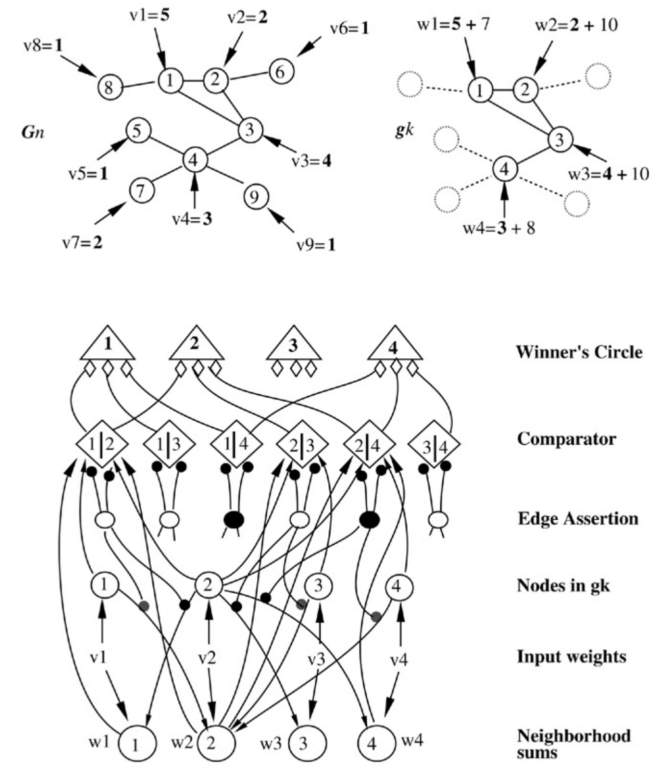

Fig. 2. In the upper left is a very simple domain model $\mathbf{G}_n$ relating nine alternatives, with input weights v_i. At the upper right is the reduced version depicted as the subgraph $\mathbf{g}_k$ with 4 nodes that correspond to the nodes in $\mathbf{G}_n$ that have the four highest maximum Borda* scores. These nodes in $\mathbf{g}_k$ continue to receive weights from nodes adjacent in $\mathbf{G}_n$, indicated by the unlabeled dashed circles. However, only nodes in $\mathbf{g}_k$ will be used in carrying out the Condorcet tally. These nodes, with their augmented weights w_i, are shown in the network below as the "neighborhood sums", which are the inputs v_i to the layer labeled "nodes in $\mathbf{g}_k$". (These neighborhood sums must be made explicit in the network.) The scheme then proceeds analogously to the Borda* network, where each node in $\mathbf{g}_k$ sends its input weight to a set of ${}_nC_2$ comparator nodes. Between these two layers is, as before, another set of ${}_nC_2$ interneurons that makes explicit through inhibition whether an edge in the domain model $\mathbf{G}_n$ is present or not. Critical is whether nodes are adjacent in $\mathbf{g}_k$ when weights are being compared. If so, as discussed in the text, the weight of the $\mathbf{g}_k$ node must be doubled. This is the role of the back-projections of the interneuron "edge assertion" layer onto the neighborhood sums (note small black circles depicting inhibitory synapses). Finally, the loser of each pair-wise comparison in the second highest layer (diamonds) sends an inhibitory signal to its namesake in the winner's circle (triangle), which shuts down this node, as it can not be the Condorcet winner. If all nodes in the winner's circle become inactivated, then there must be a top-cycle.

(loser) or "1" (winner) signal into the appropriate node in the "winner's circle".

(vii) Do a logical "AND" of the inputs to each of the k nodes in the winner's circle. If there is a unique winner, then only one node will remain active. If there is no such unique winner, then there is either a tie or a top-cycle. (See earlier definition.)

Note that although there are only k nodes in the winner's circle, in the comparator layer there will be a much larger set of roughly $2 \times {}_kC_2$ depending upon the tiling of neurons. This comparator layer, and also the comparable edge assertion layer, are the critical components that govern the size of the network. If the diameter of $\mathbf{G}_n$ is very large, the connectivities required become too distant. Some hint of this problem is given in Fig. 2 for $k = 4$. This depiction also makes clear that neither $\mathbf{G}_n$ nor $\mathbf{g}_k$ appear explicitly as graphs. Rather, the connectivity is represented by the filled nodes that indicate whether the vector [2, 1, 0] or [1, 1, 0] should be applied to the paired comparison in the comparator module. This representational form has the obvious benefit that weighted edges, i.e. correlations among alternatives, can easily be incorporated by allowing analog, rather than binary inhibition by the "edge assertion" nodes in layer 3 (small circles).

6. Success of $\mathbf{g}_k$

Fig. 3 shows the success rate of the k-Condorcet procedure for graphs of size $n < 250$, with different choices for k. The models $\mathbf{M}_n$ used were connected random graphs with edge probability 1/4. (Similar or better results are found for higher edge probabilities.) A set of weights on the nodes was chosen from a uniform distribution. Each point is based on a minimum of 100 trials, with more trials included so that the percent standard errors would be less than 5% of the ordinate values, or 0.5% if the ordinate is less than 5%. Winners were also calculated using both the Plurality (i.e. node with greatest weight) and Borda* procedures for the same set of weights. The ordinate of Fig. 3 indicates the failure rate of $\mathbf{g}_k$ to yield the same Condorcet winner as $\mathbf{G}_n$. Also shown is the behavior of the Plurality method (P) and the Borda* count, compared with the Condorcet choice. Regardless of n, the Borda* and Condorcet winners differ only about 10% of the time, as indicated by the arrow at the right. A small fraction of this percent is due to top cycles in $\mathbf{G}_n$. Likewise, an important factor for different winners for $\mathbf{g}_k$ and $\mathbf{G}_n$ is the presence of additional top cycles in $\mathbf{g}_k$. In other words, when $\mathbf{g}_k$ picks a winner, this winner is almost certainly the $\mathbf{G}_n$ winner (98+% for $k > 8$). The approximation by $\mathbf{g}_k$ is thus conservative: there are few false positives, instead no winner is chosen, unlike any Borda count.

7. Biological feasibility

Recently there has been an increasing interest in applying decision-making tools and theories from economics, political science and cognitive science to understand how neuronal systems might function as optimal decision-makers (Glimcher, 2003; Glimcher & Rustichini, 2004). Given this interest, one naturally asks whether the simple and popular Plurality-WTA network can be replaced by a more powerful information aggregation and decision-making procedure, and if so at what cost? Clearly the WTA method is very simple to implement, the Borda* next, with the Condorcet network being the most complex. Is the Condorcet network too complex? Is the additional complexity worth the benefit? Surprisingly, from a biological perspective, the Condorcet network is still rather trivial (Marr, 1969). A more interesting issue then is how the Condorcet network might actually be implemented in detail. For example, should the network be broken into overlapping modules or "receptive fields" of size k for the local calculation, but with global inputs of size n? Local tilings of receptive fields for $k > 12$ seem unlikely. But, as seen from Fig. 3, even with $k \sim 12$, similarity relations or correlations among over one hundred alternatives or events could still be evaluated quite successfully.

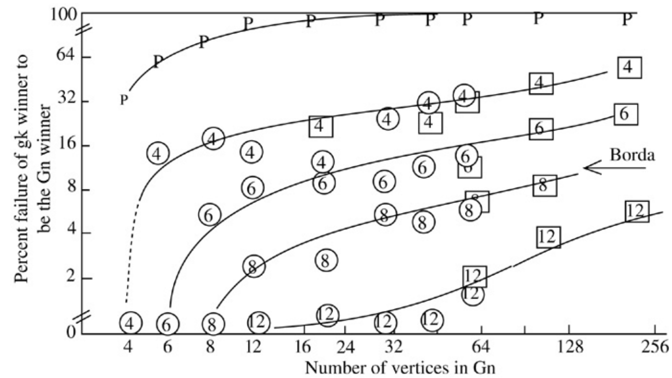

Fig. 3. Winners for $\mathbf{g}_k$ compared with winners for $\mathbf{G}_n$. The numbers along the curves indicate the values of k. Note that for $k > 6$, the smaller subgraph $\mathbf{g}_k$ does a very good job of finding the true Condorcet winner — roughly the same as if only the Borda* winner were selected (arrow at left shows the Borda* success is about 90%). However, the Borda* procedure will not reveal the top-cycle cases where there is no clear majority winner. Finally, note that the Plurality winner (P), which is the maximum weight node in $\mathbf{G}_n$, is rarely the Condorcet or Borda* winner.

The Condorcet network has a rather surprising benefit over the somewhat simpler Borda* network. Although each uses information about alternatives or similar choices, the Condorcet network explicitly finds correlations among the most significant because it must make explicit whether edges in $\mathbf{M}_n$ are adjacent or not. This design thus gives the network the potential to learn priors on such correlations. Furthermore, it has a clear rejection strategy during the learning phase, namely the presence of top-cycles. No other method mentioned above has this kind of built-in feedback mechanism because all others always output a winner. Finally, it is not inconceivable to see the potential of such a layered network design in primitive cortical areas, even for the aggregation of rather simple features.

Acknowledgements

Supported under AFOSR contract# 6894705. We gratefully acknowledge the helpful comments of two reviewers.

Appendix

A crucial step for our networks is the choice of a revised Borda rank vector that is likely to yield a true majority winner, namely one that approximates the Condorcet winner. In the absence of priors on relations among alternatives imposed by a shared domain model, Young (1995) has shown that the maximum likelihood estimate of the best social choice is either the classical Borda Count or the Condorcet tally, depending upon whether or not one seeks simply the winning alternative (Borda), or rather the entire social order (Condorcet). Obviously, in many cases the winners are the same, or occupy close positions when both social orders are compared. In our scenario with constrained preference orders, however, the classical Borda Count typically yields social outcomes and winners that depart wildly from the Condorcet winner. To clarify the difference, consider only the top three preference levels where Borda rank vectors will have the normalized form $[1, b, 0]$. In the presence of $\mathbf{M}_n$, the value of b for the classical Borda Count will depend on the vertex degree $\deg(i)$, associated with the voter's first choice a_1. Each of this same individual's second choices will then be assigned $0.5/\deg(i)$ points. Hence the total weight given to all second choices will be half the sum of the average weight of the second choices. For the same $b = 0.5$, our Borda* tally, on the other hand, would take half the sum of all the weights of the second choices. The two weighted sums will thus differ by a factor of $\deg(i)$. Never-the-less, our proposed Borda* rank vector of [1, 0.5, 0] will give the best estimate of the Condorcet winner, or equivalently the best estimate of the true majority winner and one that can not be overturned in any pairwise contest.

Fig. 4 justifies this claim. In this figure, we chose 5000 random graphs of 100 nodes, with edge probabilities of 0.1, 0.2, 0.5 and 0.9, and with weights on nodes taken from a uniform distribution. The abscissa shows the variation in the Borda* rank vector, with [1, 0, 0] corresponding to the Plurality tally, and [1, 1, 0] another extreme where second choices are given the same weight as first. Clearly the choice of [1, 0.5, 0] is the best Borda* rank vector overall, except when the graphs are very sparse.

An intuitive explanation underlying our decision to sum over all second choice weights is that whenever alternatives lie equidistant from two other alternatives being compared in a Condorcet tally, then shared weights will cancel, as they should (see Richards et al., 2002). This would not be the case if weights were adjusted depending on vertex degrees.

Related to the above choice of the Borda* rank vector is the agreement between Borda* and Condorcet winners when the order of the social outcomes is considered. From Fig. 4, first note that if $\mathbf{M}_n$ has the form of a random graph, then with edge probability greater than 0.1, and with the rank vector [1, 0.5, 0], over 89% of the Condorcet winners will agree with the Borda* winner. For that residual percent where the winners do not agree, simulations show that about 70% of the remainder (i.e. 7% of the total) of the Condorcet winners will equal the Borda* runner-up. More importantly, if $n > 20$, the Condorcet winner is almost always in the top five of the Borda* social order, with the likelihood increasing as n increases. Hence $\mathbf{g}_k$ even for $k = 12$ is almost surely guaranteed to contain the Condorcet winner for $\mathbf{G}_n$.

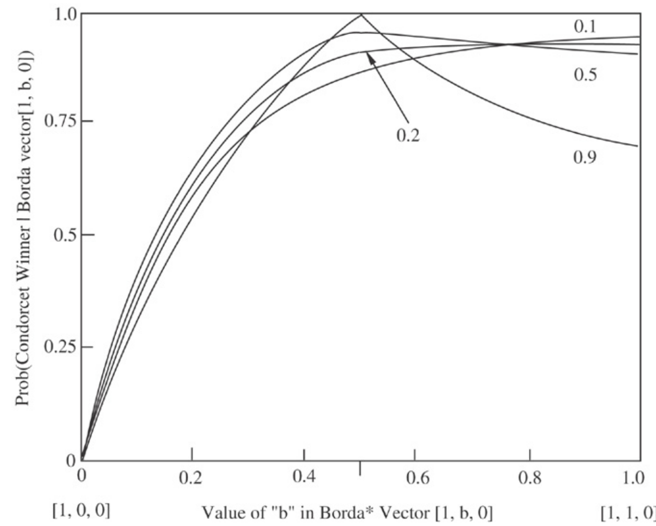

Fig. 4. For $\mathbf{M}_n$ modeled as a random graph, with weights on nodes chosen from a uniform distribution, the Borda* rank vector [1, 0.5, 0] provides the best overall choice when the Borda* winner is used to predict the Condorcet winner. Each curve is based on sampling 5000 random graphs with 100 nodes, and the result smoothed. Edge probabilities were 0.1, 0.2, 0.5, and 0.9. The results are representative of connected random graphs for $n > 20$. At [1, 0.5, 0], the ordinate values are respectively 0.89, 0.93, 0.97 and 0.999.

References

Amari, S., & Arbib, M. A. (1977). Competition and cooperation in neural nets. In J. Metzler (Ed.), *Systems neuroscience* (pp. 119–165). New York: Academic Press.

Arrow, K. J. (1963). *Social choice and individual values*. New York: Wiley.

Borda, J. -C. (1784). Memoire sur les elections au Scrutin. Histoire de l'Academie Royal des Sciences.

de Condorcet, M. (1785). Essai sur l'application de l'analyse a la probabilite des decisions rendue a la pluralite des voix, Paris (see Arrow, 1963).

Dodgson, C. (1876). A method of taking votes on more than two issues. In D. Black (Ed.), *The theory of committees and elections* (pp. 222–234). Cambridge, UK: Cambridge University Press (1958).

Glimcher, P. W. (2003). *Decisions, uncertainty and the brain: The science of neuroeconomics*. Cambridge, MA: MIT Press.

Glimcher, P. W., & Rustichini, A. (2004). Neuroeconomics: The consilience of brain and decision. *Science*, *306*, 447–452.

Harary, F. (1969). *Graph theory*. Reading, MA: Addison-Wesley.

Klamler, C. (2004). The Dodgson ranking and the Borda Count: A binary comparison. *Mathematical Social Sciences*, *48*, 103–108.

Maass, W. (2000). On the computational power of winner-take-all. *Neural Computation*, *12*, 2519–2636.

Marr, D. (1969). A theory of cerebellar cortex. *Journal of Physiolology (London)*, *202*, 437–470.

Ratliff, T. C. (2002). A comparison of Dodgson's method and the Borda Count. *Economic Theory*, *20*, 357–372.

Richards, D. (2001). Coordination and shared mental models. *American Journal of Political Science*, *45*, 250–276.

Richards, W. (2005). Collective choice with uncertain domain models. AI-Memo 2005-54. Available at publications.csail.mit.edu/tmp/mit-CSAIL-TR-2005-054.

Richards, D., McKay, B., & Richards, W. (1998). Collective choice and mutual knowledge structures. *Advances in Complex Systems*, *1*, 221–236.

Richards, W., McKay, B., & Richards, D. (2002). Probability of collective choice with shared knowledge structures. *Journal of Mathematical Psychology*, *46*, 338–351.

Runkel, P. J. (1956). Cognitive similarity in facilitating communication. *Sociometry*, *19*, 178–191.

Saari, D. G. (1994). *Geometry of voting*. Berlin: Springer-Verlag.

Saari, D., & Haunsberger, D. (1991). The lack of consistency for statistical decision procedures. *The American Statistician*, *45*, 252–255.

Xie, X. -H., Hahnloser, R., & Seung, H. S. (2001). Learning winner-take-all competitions between groups of neurons in lateral inhibiting networks. In *Advances in neural information processing*: *Vol. 13* (pp. 350–356). Cambridge, MA: MIT Press.

Young, H. P. (1995). Optimal voting rules. *Journal of Economic Perspectives*, *9*, 51–64.