ELSEVIER

Discrete Applied Mathematics 69 (1996) 33–60

DISCRETE APPLIED MATHEMATICS

Representation theory and invariant neural networks ☆

Jeffrey Wood *, John Shawe-Taylor

Department of Computer Science, Royal Holloway, University of London, Egham, Surrey TW20 0EX, UK

Received 5 April 1994; revised 19 December 1994

Abstract

A feedforward neural network is a computational device used for pattern recognition. In many recognition problems, certain transformations exist which, when applied to a pattern, leave its classification unchanged. Invariance under a given group of transformations is therefore typically a desirable property of pattern classifiers. In this paper, we present a methodology, based on representation theory, for the construction of a neural network invariant under any given finite linear group. Such networks show improved generalization abilities and may also learn faster than corresponding networks without in-built invariance.

We hope in the future to generalize this theory to approximate invariance under continuous groups.

1. Introduction

A typical property of a pattern recognition problem is the invariance of the classification under certain transformations of the input pattern. By constructing the pattern recognition system in such a way that the invariance is in-built a priori, it should be possible to speed training and/or improve the generalization performance of the system.

Numerous papers have been written on the subject of invariant pattern recognition (see for example [6, 10, 17, 19]), but few make use of the wealth of highly applicable material in the field of group theory. In this paper we apply standard representation theory to obtain a number of interesting and general results.

For certain finite groups, namely permutation groups, invariance can be achieved by building a Symmetry Network (see [13, 15]) in which connections share weights. This work generalizes the theory of Symmetry Networks to the problem of invariance under any given finite linear group.

☆ This research was funded by the Engineering and Physical Sciences Research Council and the Defence Research Agency under CASE award no. 92567042.

* Corresponding author.

SSDI 0166-218X(95)00075-5

We begin by presenting the necessary background in the field of neural networks. We then summarize the theory of Symmetry Networks and introduce some representation-theoretical notation. In Section 2 we introduce a new class of networks, Group Representation Networks, and give a classification of group representations based on a property which amounts to their degree of applicability in these networks. In Section 3 we consider a special class of Group Representation Networks, namely Permutation Representation Networks. In Section 4 we turn to other classes of Group Representation Networks. In Section 6 we discuss some practical issues and summarize the results of some simulations. In Section 7 we conclude.

This paper is based on standard group representation theory. The results we use can be found in most introductory texts on the subject, see for example [1,2,4].

1.1. Feedforward networks

In this section we summarize basic neural network terminology. For further information, see the classic literature on the subject [7,8,11] or for an overall view of the field see [3,5].

A *feedforward neural network* is a (finite) connected acyclic directed graph with certain properties. Each edge (*connection*) of the network has an associated *weight*, which is usually a real number. Each node (or *neuron*) of the network has an associated function, the *activation function* which is a mapping from the real numbers to some subset of the real numbers.

The *inputs* or *input nodes* of the network are those neurons with no connection leading into them. The input nodes have no activation functions, or can be regarded as having the identity function as an activation function. The *outputs* or *output nodes* of the network are those neurons with no connection leading from them. The remaining nodes are known as *hidden nodes*.

Neurons with the same connectivity are typically arranged in a *layer*. The input nodes form one layer and the output nodes another. There may be one or more *hidden layers* (layers of hidden nodes).

Fig. 1. shows a typical network, in which the neurons are grouped together in layers. We have not fully connected those layers which have connections between them, since this would have cluttered the diagram. The information flow in the network is from the bottom of the figure upwards. This is a conventional manner of representation of a network. Hence we refer to layer A as being *higher than* layer B if information flows from B to A (though not necessarily directly). The output layer is clearly the highest layer and the input layer the lowest. The functionality of the network is defined as follows. The input of the network is a vector of real values which defines the output of each input node. The output of the network is the vector of values which are the outputs of each output node. The network's output is computed as follows .

The *net input* of a given hidden or output node j is the weighted sum $\sum_i w_{ji}x_i$, where the summation is done over all nodes i with connections leading to node j. x_i denotes the output of node i and w_{ji} denotes the weight of the connection from node i

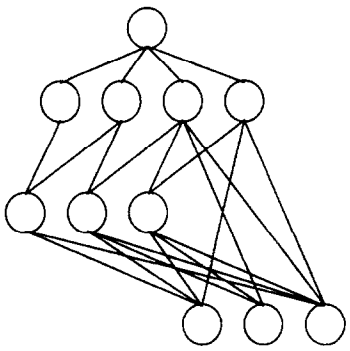

Fig. 1. A feedforward network

to node j. The output of node j is then determined by applying its activation function f_j, i.e.

$$x_j = f_j\left(\sum_i w_{ji}x_i\right).$$

f_j is typically taken to be the logistic function $f_j(x) = 1/1 + exp(-x)$, though we make no such assumptions in this paper.

The output of the nodes is calculated in strict graph traversal order (i.e. no node's output is calculated until that of all the nodes with connections leading into it have been calculated), finishing with the output nodes.

Usually each hidden and output node also has a *threshold* value associated with it, which is subtracted from the weighted sum $\sum_i w_{ji}x_i$ before the activation function f_j is applied. In this paper we treat this operation of subtraction of a threshold value as being part of the activation function. No generality is lost by this assumption.

The outputs or net inputs of a given layer can be represented by a vector. The contribution of the outputs of one layer to the net inputs of a directly connected higher layer is a linear transform. It can therefore be represented by a *weight matrix* W of weights w_{ji}.

Network training

A network can be trained, by use of a *learning algorithm*, to emulate a given function. This is done by presenting the network with a set of example inputs and corresponding desired output values. The learning algorithm then iteratively adapts the weights of the network's connections so as to minimize the difference between the desired and actual network outputs. This is typically done by means of a gradient descent method, such as the classical backpropagation algorithm introduced in [12].

If a network succeeds in correctly classifying a training set, then a test can be made of its *generalization ability*. This is the ability of the network to correctly classify inputs which were not presented to it during the operation of the learning algorithm.

A network which fails to generalize well has not learned the target concept correctly, but may succeed if further trained on a larger set of examples.

1.2. Symmetry Networks

The concept of a Symmetry Network was introduced in [13]. Essentially, a Symmetry Network is a feedforward neural network which is invariant under a group of permutations of the input.

This invariance is achieved by extending each permutation from its natural action on the input nodes to an *automorphism* of the network, i.e. a permutation of all network nodes. Each automorphism has an obvious induced action on the connections of the network. An automorphism is said to be *weight-preserving* if the weight of any two connections in the same orbit is the same.

Formally, a *Symmetry Network* is a pair $(\mathcal{N}, \mathcal{G})$, where $\mathcal{N}$ is a feedforward network and $\mathcal{G}$ a group of weight-preserving automorphisms of $\mathcal{N}$ which are required to fix each output node. In addition we require that the activation function is fixed over each node orbit (this is not a stringent requirement since it is usual to fix the activation function over an entire network).

The output of a Symmetry Network is guaranteed to be an invariant under the action of $\mathcal{G}$ on the input nodes.

Since the group of automorphisms is weight-preserving, any two nodes in a given orbit have equivalent, if not identical, connectivities. Thus for convenience we associate node orbits with network layers, although the correspondence is a little misleading. In particular with this association each output node appears in a separate layer.

With each orbit (layer) we can associate a subgroup, namely the stabilizer of an arbitrarily chosen node of that orbit. The nodes of the orbit are then naturally associated with that subgroup's cosets, the number of nodes in the layer being equal to the subgroup's index. This suggests a natural way to construct a Symmetry Network under a given group $\mathcal{G}$; we choose a subgroup for each hidden layer. The subgroup corresponding to the input nodes is determined by the action of $\mathcal{G}$ on the input nodes, which is given by the problem in hand. The subgroup corresponding to each output node is the whole group $\mathcal{G}$.

The connections between two given network layers are partitioned into orbits, two connections in the same orbit being constrained to have equal weight. In [18] it is shown that the orbits of network connections between layers associated with subgroups $\mathcal{F}$ and $\mathcal{H}$ correspond precisely to the double cosets of the pair $(\mathcal{F}, \mathcal{H})$. Furthermore, the correspondence is specifiable, which allows a relatively efficient construction of the connection orbit structure.

Conventional learning algorithms can readily be adapted and applied to Symmetry Networks. The only change in such algorithms is that the variable parameters of the network are now the weights of the connection orbits, rather than the weights of the connections themselves.

One open issue in the subject of Symmetry Networks is that of *discriminability*, i.e. the ability of the network to discriminate between input patterns which are not in the same orbit under the invariance group. This issue is discussed in the paper [15], where examples of both fully discriminating and partially discriminating networks are given, with reference to the graph isomorphism problem.

1.3. Definitions and notation from representation theory

In this section we list some terminology and notation which we will use from the field of representation theory.

We always denote our finite invariance group by $\mathcal{G}$, with $\mathcal{H}$ and $\mathcal{F}$ denoting subgroups. We use A, B and M to refer to completely arbitrary group representations. P will be used to denote an arbitrary permutation representation and Z an arbitrary *inversion representation* (defined below).

Given a representation A, we write $\chi(A)$ for its character. We denote the inner product on characters over a group $\mathcal{G}$ or subgroup $\mathcal{H}$ by $\langle , \rangle_{\mathcal{G}}$ and $\langle , \rangle_{\mathcal{H}}$ respectively.

Given representations A and B of $\mathcal{G}$, we denote by $A \otimes B$ the direct product of A and B.

We refer to a homomorphism from representation A to representation B as meaning a homomorphism from the (left) module under $\mathcal{G}$ associated with A to that associated with B. All representations dealt with in this paper are assumed to be associated with left modules. This is an arbitrary decision, consistency being the only requirement.

The space of homomorphisms from a module U to a module V is called the *intertwining space* of (U, V) and its dimension the *intertwining number* of U and V. Since we deal with representations, rather than modules, we will refer instead to the intertwining space and number of the corresponding representations.

For a representation A of a subgroup $\mathcal{H}$ of $\mathcal{G}$, we denote by $\mathrm{ind}_{\mathcal{H}}^{\mathcal{G}} A$ the induced representation of $\mathcal{G}$ from A. For a representation B of $\mathcal{G}$ and given a subgroup $\mathcal{H}$ of $\mathcal{G}$, we denote by $\mathrm{res}_{\mathcal{H}}^{\mathcal{G}} B$ the restricted representation of B to $\mathcal{H}$.

Let $\mathcal{H}$ be a subgroup of $\mathcal{G}$. The *fixed point subspace* of $\mathcal{H}$ (in a given module V under $\mathcal{G}$) is the space of all vectors of V which are fixed by every element of $\mathcal{H}$.

Finally we wish to make a non-standard definition. An *inversion matrix* or reflection matrix is a diagonal matrix with diagonal entries of ± 1. An *inversion representation* is a matrix representation in which all group representative matrices can be written as the product of an inversion matrix and a permutation matrix.

2. Group Representation Networks

We have said that each layer of a Symmetry Network is associated with a subgroup of the invariance group $\mathcal{G}$. The action of $\mathcal{G}$ on the nodes of a layer is then equivalent to its action on the cosets of the corresponding subgroup; i.e. the subgroup defines a permutation representation of the invariance group.

It is possible to associate layers of the network with group representations which are not permutations. This is a particularly useful technique to apply to the input layer since it allows the construction of a network invariant under an arbitrary (finite) group of linear transformations, as will be seen shortly.

We now formalize this concept, which requires a preliminary definition. We write $\underline{f}$ to denote the extension of $f : \mathbb{R} \mapsto \mathbb{R}$ to action on a vector of arbitrary dimension (in the obvious componentwise manner).

Definition 2.1. Let A be an n-dimensional matrix representation of the group $\mathcal{G}$. Then the function $f : \mathbb{R} \mapsto \mathbb{R}$ is said to *preserve the representation* A if

$$\underline{f}(A(g)v) = A(g)\underline{f}(v) \quad \forall v \in \mathbb{R}^n, \ g \in \mathcal{G}$$

Definition 2.2. A *Group Representation Network (GRN)* is a pair $(\mathcal{N}, \mathcal{G})$, where $\mathcal{N}$ is a feedforward network and the following conditions apply :

1. The nodes are partitioned into layers. With each layer is associated a representation of $\mathcal{G}$, of dimension equal to the number of nodes in the layer. There are no intra-layer connections.

2. Each output node is in a layer by itself, associated with the trivial representation. The input nodes also form one or more layers.

3. The matrix of weights from a layer of nodes corresponding to representation A to a layer corresponding to representation B is a homomorphism from A to B.

4. The activation function is fixed over a given layer and preserves the group representation associated with that layer.

A Symmetry Network is a special case of a Group Representation Network in which every layer's associated representation is a permutation representation.

We now state and prove the result which forms the basis of this paper.

Theorem 2.3. *The output of a GRN* $(\mathcal{N}, \mathcal{G})$ *is invariant under the action of* $\mathcal{G}$ *on the input layer of* $\mathcal{N}$.

Proof. We start by indexing the layers of $\mathcal{N}$ by $0, 1, \ldots L$, on the understanding that there are no connections *from* a layer *to* any lower-indexed layer. Hence layer 0 is the input layer and the output layers are those with the highest indices.

Let A_i denote the matrix representation associated with layer i, f_i the activation function of layer i and $y_i(x)$ the vector of outputs from the nodes of layer i given x as input to the whole network.

Let $W(i,j)$ denote the matrix of weights of the connections leading from layer i to layer j (thus $j > i$). $W(i,j)$ will be the zero matrix for disconnected layers.

Now consider two network inputs x_1 and x_2 which are related by a group transformation $g \in \mathcal{G}$. Let $P(i)$ be the statement

$$A_i(g)y_i(x_1) = y_i(x_2)$$

We now prove $P(i)$ for $i \in 0 \ldots L$ by induction on i.

$P(0)$ is true since $y_0(x_1) = x_1$ and $y_0(x_2) = x_2$ are the inputs of the network which are related by transformation g according to the representation A_0. Assume $P(0) \ldots P(k)$ are true; thus $A_i(g)y_i(x_1) = y_i(x_2)$ for all $i \in 0 \ldots k$. We now have:

$$
\begin{aligned}
y_{k+1}(x_2) &= \underline{f_{k+1}}\left(\sum_{i=0}^{k} W(i,k+1)y_i(x_2)\right) \quad \text{(this is the functionality of the network)} \\
&= \underline{f_{k+1}}\left(\sum_{i=0}^{k} W(i,k+1)A_i(g)y_i(x_1)\right) \quad \text{(by induction hypothesis)} \\
&= \underline{f_{k+1}}\left(A_{k+1}(g)\sum_{i=0}^{k} W(i,k+1)y_i(x_1)\right) \\
&\quad (W(i,k+1) \text{ is a homomorphism from the representation } A_i \text{ to } A_{k+1}) \\
&= A_{k+1}(g)\underline{f_{k+1}}\left(\sum_{i=0}^{k} W(i,k+1)y_i(x_1)\right) \quad (\text{since } f_{k+1} \text{ preserves} A_{k+1}) \\
&= A_{k+1}(g)y_{k+1}(x_1).
\end{aligned}
$$

Hence $P(k+1)$ is true.

By the principle of induction, $P(L')$ is true for all output layers L', i.e.

$$y_{L'}(x_2) = A_{L'}(g)y_{L'}(x_1).$$

However, each output layer corresponds to the trivial representation, so $y_{L'}(x_2) = y_{L'}(x_1)$ for all output layers L'. Hence any two inputs in the same orbit under $\mathcal{G}$ yield the same output of $\mathcal{N}$. □

We are now in a position to investigate the use of different representations in a Group Representation Network. This requires some preliminary notation: For any real number a, we define the following functions;

$$
l_a^+, l_a^- : \mathbb{R} \mapsto \mathbb{R}, \text{by } l_a^+(x) = \begin{cases} ax, & x \geqslant 0, \\ \text{undefined}, & x < 0, \end{cases} \quad l_a^-(x) = \begin{cases} \text{undefined}, & x > 0, \\ ax, & x \leqslant 0. \end{cases}
$$

Now we define a set of 'nicely-behaved' functions Φ as follows:

$\Phi = \{\phi : \mathbb{R} \mapsto \mathbb{R} \mid \phi$ intersects l_a^+ at an infinite number of places for at most one value of a, and ϕ intersects l_a^- at an infinite number of places for at most one value of $a\}$.

The following theorem provides a classification of all group representations according to the functions in Φ which preserve those representations.

Theorem 2.4. *Let A be a finite-dimensional representation of the group $\mathcal{G}$ acting on a real vector space, $f : \mathbb{R} \mapsto \mathbb{R}$ a function which is such that $f_0 : \mathbb{R} \mapsto \mathbb{R}$ defined by $f_0(x) = f(x) - f(0)$ is in Φ. Then f preserves A if and only if one of the following conditions holds :*

1. *A is a permutation representation.*

2. *A is an inversion representation and f is odd.*

3. *A is a unit-row representation (i.e. every row of every matrix of A sums to* 1*) and f is affine.*

4. *A is a positive representation (i.e. every entry of every matrix of A is non-negative) and f is semilinear (i.e. there exist reals* k_1 *and* k_2 *such that* $f(x) = k_1 x$ *for* $x \geqslant 0$, $f(x) = k_2 x$ *for* $x < 0$*).*

5. *f is linear.*

The proof of this theorem appears in Appendix A. We note that most functions of practical interest (with a few exceptions such as $\tan x$) are in Φ.

Theorem 2.4 effectively provides a categorization of group representations into classes which are preserved by different types of activation function. The reader should note that, since subtraction of a threshold is being regarded as part of the activation function, thresholds cannot be used in neuron layers corresponding to representations other than permutations and unit-row representations.

In Sections 3 and 4 we consider the use in a GRN of each of our named classes of representation. Before that, however, we give some general results on the structure of a single layer of connections in a GRN.

2.1. Weight matrix form

We now consider the structure of the connections between a given pair of layers in a Group Representation Network. All the information about these connections is contained in the weight matrix W, which is the linear transformation that yields the net input of the higher layer given the output of the lower layer. By the definition of a GRN, W is a homomorphism from the representation of $\mathcal{G}$ associated with the lower layer to that associated with the higher layer. Thus W is an element of the intertwining space of the pair of representations concerned. We now state an explicit formula for any weight matrix W.

Lemma 2.5. *Let A and B be two representations of group* $\mathcal{G}$ *(corresponding to left modules). Then any homomorphism W from A to B is of the form*

$$W = \frac{1}{|G|} \sum_{g \in \mathcal{G}} B(g) X A(g^{-1})$$

for a fixed matrix X, and furthermore any matrix of this form is a homomorphism from A to B.

Proof. Let W be a homomorphism from A to B. Hence

$$WA(h) = B(h)W$$

for all $h \in \mathcal{G}$. Hence we find that $X = W$ satisfies the equation of the lemma, so W is of the required form.

Conversely, let W be of the form in the lemma. Now for all $h \in \mathcal{G}$,

$$\begin{aligned} WA(h) &= \frac{1}{|G|} \sum_{g \in \mathcal{G}} B(g) X A(g^{-1}h) \\ &= \frac{1}{|G|} \sum_{g' \in \mathcal{G}} B(hg') X A(g'^{-1}) \quad \text{(reordering the sum by setting } g' = h^{-1}g\text{)} \\ &= B(h)W. \end{aligned}$$

Hence W is a homomorphism from A to B. □

From Schur's Lemma we immediately derive the following results for the special case for when the representations we are concerned with are irreducible.

Lemma 2.6. *Let A and B denote two representations corresponding to connected layers of a Group Representation Network. Then we have the following results for any weight matrix W of the connections between the two layers.*

1. *If A and B are irreducible and equivalent then $W = kI$ for some $k \in \mathbb{R}$.*
2. *If A and B are irreducible and inequivalent then $W = 0$.*

Lemma 2.5 gives a formula for any valid weight matrix W. By using a matrix X with independent variables as entries we obtain a completely general weight matrix W in which each entry is a linear combination of the entries of X. We call this matrix the *generalized weight matrix* of the pair (A, B) and denote it by $W_{A,B}$. A specific weight matrix is given by any mapping of the variables in X to the real numbers. Henceforth when we refer to a weight matrix we will be referring only to a generalized matrix as defined in this manner. Note that $W_{A,B}$ defines and is defined by the intertwining space of (A, B).

Whilst the entries of X are linearly independent, the entries of $W_{A,B}$ will almost always be expressible in terms of a smaller set of parameters. The minimum number of parameters in terms of which the general weight matrix $W_{A,B}$ can be expressed is the number of linearly independent weights. We refer to this number as the *parameter dimension* $p(A, B)$ of $W_{A,B}$. Hence $p(A, B)$ is the intertwining number of A and B.

A standard result of representation theory is that the intertwining number of representations A and B is equal to the inner product of the characters of A and B (see for example [1]). Thus for our application of representation theory we obtain:

Lemma 2.7. *Given any two representations A and B of $\mathcal{G}$, we have*

$$p(A, B) = \langle \chi(A), \chi(B) \rangle_{\mathcal{G}}.$$

Another standard result is that the intertwining number of two permutation representations is the number of double cosets of the pair of subgroups corresponding to those representations. This explains the connection between the connection orbits and double cosets in a Symmetry Network.

2.1.1. Direct product representation form

The formula of Lemma 2.5 expresses the weights as linear combinations of a set of parameters. However this is not a very useful form for the formula. We rewrite it in terms of the direct product of the representations A and B, this being the representation of $\mathscr{G}$ defined by $\mathscr{G}$'s action on the network connections. To do this, we require a result to do with the direct product of arbitrary matrices.

For an arbitrary m by n matrix X, let $c(X)$ denote the mn-dimensional vector obtained by writing the columns of X out one after another, to form a column vector. Similarly, let $r(X)$ denote the vector obtained by writing out the rows of X in a column vector (not a row vector).

Lemma 2.8. *Given $W = BXA$ where all matrices are arbitrary, we have*

1. $c(W) = (A^{\mathrm{T}} \otimes B)c(X)$,
2. $r(W) = (B \otimes A^{\mathrm{T}})r(X)$.

Here $^{\mathrm{T}}$ denotes transpose and $\otimes$ denotes direct product.

The proof of this result is lengthy but relatively simple and of no interest in its own right. We therefore leave it to the reader.

We now apply this result to the formula of Lemma 2.5. We first assume that the representations A and B are orthogonal; thus $A(g)^{\mathrm{T}} = A(g^{-1})$ for all elements g of $\mathscr{G}$. The relationship we get is

$$r(W_{A,B}) = \frac{1}{|\mathscr{G}|}\left(\sum_{g\in\mathscr{G}} B(g)\otimes A(g)\right) r(X).$$

We now introduce a new matrix $\Phi_{A,B}$ by the formula

$$r(W_{A,B}) = \Phi_{A,B} r(X), \quad \Phi_{A,B} = \frac{1}{|G|}\sum_{g\in\mathscr{G}} B(g)\otimes A(g). \tag{1}$$

We call the matrix $\Phi_{A,B}$ the *coefficient matrix* of the representations A and B. It defines a mapping from the space of possible parameter matrices X to the intertwining space of (A,B). $\Phi_{A,B}$ can also be regarded as the mean value of the direct product representation $A \otimes B$, which as a group average is naturally invariant under the action of any group element.

The trace of $\Phi_{A,B}$ is the average character of the representation $A \otimes B$ over the whole group. Since the character of $A \otimes B$ is equal to the product of the characters of A and B, this trace is equal to $\langle \chi(A), \chi(B)\rangle$, which by Lemma 2.7 is the parameter dimension of $W_{A,B}$.

As remarked earlier, the parameter dimension of $W_{A,B}$ is the number of linearly independent weights, which is clearly the rank of $\Phi_{A,B}$. Thus we have:

Lemma 2.9. *The rank and trace of the coefficient matrix $\Phi_{A,B}$ are equal to $p(A,B)$.*

That the rank of $\Phi_{A,B}$ is equal to its trace is inevitable since $\Phi_{A,B}$ is in fact a projection matrix.

Note however that unless the coefficient matrix has full rank (which will not usually be the case), there will be redundant parameters. It would be useful to find an expression for the weights in terms of a minimal set of parameters.

This can be done by column-reducing the matrix $\Phi_{A,B}$ to Echelon form, which we now define (following the definition given in [9]): Let A be an m by n matrix; let r_i denote the ith row of A which is not a linear combination of preceding rows. Then A is said to be in *column-reduced Echelon form* if r_i is equal to the ith row of the n by n identity matrix for all $i \in 1 \ldots \text{rank}(A)$. We define row-reduced Echelon form in an analogous manner.

Now given any invertible matrix E (in particular an elementary matrix), $E^{-1}r(X)$ is still a vector of independent parameters. Thus we can write

$$r(W) = \overline{\Phi_{A,B}}x,$$

where $x = E_1^{-1}E_2^{-1} \ldots E_k^{-1}r(X)$ is a parameter vector and $\overline{\Phi_{A,B}} = \Phi_{A,B}E_k \ldots E_2E_1$ is the column-reduced Echelon form of $\Phi_{A,B}$. This effectively expresses $r(W_{A,B})$ in terms of a parameter set with size equal to $p(A,B)$.

3. Permutation Representation Networks

We now introduce a subclass of Group Representation Networks for which we can simplify the results already presented.

Definition 3.1. A *Permutation Representation Network (PRN)* is a Group Representation Network in which every hidden layer (and trivially every output layer) corresponds to a permutation representation.

Thus all Symmetry Networks are PRNs. The converse however is not true since a PRN allows a non-permutation representation at the input layer. However the subnetwork of a PRN formed by deleting the input layer and all connections leading from it has the structure of a Symmetry Network.

Since we allow any linear representation to act on the input layer of a PRN, we can build a PRN which is invariant under any finite linear group. The use of PRNs rather than general Group Representation Networks therefore restricts the choice of network structure but not the class of groups under which invariance can be achieved. Also, recall from Theorem 2.4 that the use of permutation representations places no restriction upon the activation functions that we can use.

For a PRN, it is relatively easy to build upon the theory presented so far. Again, as in the case of a Symmetry Network, the hidden and output layers now correspond not only to representations of the invariance group $\mathcal{G}$ but also to subgroups.

3.1. Further notation and definitions

We therefore consider a given pair of layers corresponding to representations M and P. Again we take these representations to correspond to left modules under $\mathcal{G}$. P corresponds to the higher layer and hence is a permutation representation. We furthermore assume that it is transitive; if this is not the case then the higher layer can be split up into distinct layers which do correspond to transitive permutation representations, and these layers can be considered separately. The stabilizer of a node (we arbitrarily choose the first node) in the higher layer is a subgroup of $\mathcal{G}$ which we denote by $\mathcal{H}$. Let m denote the index of $\mathcal{H}$ in $\mathcal{G}$, and $h_1, h_2, \ldots, h_m$ the right coset representatives of $\mathcal{H}$ (we take $h_1 = e$).

M may or may not be a permutation representation; however we assume as before that it is orthogonal. Let n denote the dimension of representation M.

The action of $\mathcal{G}$ on the nodes of the higher layer corresponds to its action on the right cosets of $\mathcal{H}$; thus

$$P(g)_{(i,j)} = 1 \quad \Leftrightarrow \quad \mathcal{H}h_i g = \mathcal{H}h_j. \tag{2}$$

Finally, we will find the following definition very useful.

Definition 3.2. Given a representation M of a group $\mathcal{G}$ with subgroup $\mathcal{H}$, we define the *characteristic matrix* $\mathcal{M}(\mathcal{H})$ of the subgroup $\mathcal{H}$ (in the representation M) by the equation

$$\mathcal{M}(\mathcal{H}) = \frac{1}{|\mathcal{H}|} \sum_{h \in \mathcal{H}} M(h).$$

3.2. Reducing the parameter set

From Eq. (1) we have that

$$r(W_{M,P}) = \begin{pmatrix} \phi_{11} & \phi_{12} & \cdots & \phi_{1m} \\ \phi_{21} & \phi_{22} & \cdots & \phi_{2m} \\ \vdots & \vdots & & \vdots \\ \phi_{m1} & \phi_{m2} & \cdots & \phi_{mm} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{pmatrix},$$

where the ϕ_{ij}'s are submatrices of $\Phi_{M,P}$ defined by

$$\phi_{ij} = \frac{1}{|\mathcal{G}|} \sum_{g \in \mathcal{G}} P(g)_{(i,j)} M(g)$$

and $r(X)^{\mathrm{T}} = (x_1^{\mathrm{T}} x_2^{\mathrm{T}} \ldots x_m^{\mathrm{T}})$, the x_i's being n-dimensional.

By Eq. (2) this becomes

$$\begin{aligned}\phi_{ij} &= \frac{1}{|G|}\sum_{g\in h_i^{-1}\mathscr{H}h_j} M(g),\\ \phi_{ij} &= \frac{1}{m}M(h_i^{-1})\mathscr{M}(\mathscr{H})M(h_j) = \phi_{i1}M(h_j).\end{aligned} \tag{3}$$

Hence we can rewrite $r(W_{M,P})$ as

$$r(W_{M,P}) = \begin{pmatrix} \phi_{11} & 0 & \cdots & 0 \\ \phi_{21} & 0 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ \phi_{m1} & 0 & \cdots & 0 \end{pmatrix} \begin{pmatrix} I & \cdots & I \\ I & \cdots & I \\ \vdots & \vdots & \vdots \\ I & \cdots & I \end{pmatrix} \begin{pmatrix} M(h_1) & & & \\ & M(h_2) & & \\ & & \ddots & \\ & & & M(h_m) \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{pmatrix}$$

$$= \begin{pmatrix} m\phi_{11} & 0 & \cdots & 0 \\ m\phi_{21} & 0 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ m\phi_{m1} & 0 & \cdots & 0 \end{pmatrix} \begin{pmatrix} \rho^{\mathrm{T}} \\ \rho^{\mathrm{T}} \\ \vdots \\ \rho^{\mathrm{T}} \end{pmatrix} = \begin{pmatrix} m\phi_{11} \\ m\phi_{21} \\ \vdots \\ m\phi_{m1} \end{pmatrix} \rho^{\mathrm{T}},$$

where ρ is the n-dimensional parameter vector defined by $\rho^{\mathrm{T}} = 1/m\sum_{j=1}^{m} M(h_j)x_j$. Note that the entries of ρ are linearly independent.

By Eq. (3) we get

$$r(W_{M,P}) = \begin{pmatrix} M(h_1^{-1}) \\ M(h_2^{-1}) \\ \vdots \\ M(h_m^{-1}) \end{pmatrix} \mathscr{M}(\mathscr{H})\rho^{\mathrm{T}}$$

$$\Rightarrow \quad r(W_{M,P})^{\mathrm{T}} = \rho\mathscr{M}(\mathscr{H})(M(h_1)M(h_2)\cdots M(h_m)), \tag{4}$$

since M is orthogonal, and hence $\mathscr{M}(\mathscr{H})$ is symmetric.

We now present our main result on the structure of PRNs.

Theorem 3.3. *Let P be a permutation representation of $\mathscr{G}$ with $\mathscr{H}$ the subgroup stabilizing a single point. Denote by $h_1, h_2, \ldots, h_m$ the coset representatives of $\mathscr{H}$. Let M be any orthogonal representation of $\mathscr{G}$.*

Then $p(M,P)$ is equal to the trace of the characteristic matrix $\mathscr{M}(\mathscr{H})$ of the subgroup $\mathscr{H}$ in the representation M, and the weight matrix of the pair (M,P) is given by

$$W_{M,P} = \begin{pmatrix} \rho\mathscr{M}(\mathscr{H})M(h_1) \\ \rho\mathscr{M}(\mathscr{H})M(h_2) \\ \vdots \\ \rho\mathscr{M}(\mathscr{H})M(h_m) \end{pmatrix},$$

where ρ denotes an n-dimensional parameter vector.

Proof. The formula for the weight matrix follows immediately from Eq. 4. We therefore have only to prove that $p(M,P) = \text{Trace}(\mathcal{M}(\mathcal{H}))$. By Lemma 2.7 we have

$$\begin{aligned} p(M,P) &= \langle \chi(M), \chi(P) \rangle_{\mathcal{G}} \\ &= \langle \chi(M), \chi(\text{ind}_{\mathcal{H}}^{\mathcal{G}} 1) \rangle_{\mathcal{G}}, \end{aligned}$$

where 1 denotes the trivial representation of $\mathcal{H}$. That $P = \text{ind}_{\mathcal{H}}^{\mathcal{G}} 1$ is a standard result. By Frobenius' Reciprocity Theorem we get

$$\begin{aligned} p(M,P) &= \langle \chi(\text{res}_{\mathcal{H}}^{\mathcal{G}} M), \chi(1) \rangle_{\mathcal{H}} \\ &= \text{Trace}(\mathcal{M}(\mathcal{H})). \quad \square \end{aligned}$$

Thus we have written $W_{M,P}$ in terms of a parameter vector ρ, the characteristic matrix $\mathcal{M}(\mathcal{H})$ and a set of coset representative matrices. Note however that whilst the n entries of ρ are linearly independent, the parameter dimension need not be n.

3.3. *Properties of the characteristic matrix*

We now analyze the characteristic matrix of a given subgroup $\mathcal{H}$. This matrix is evidently of considerable importance since it determines the entire structure of the weight matrix.

Theorem 3.4. *Let M be an orthogonal representation of the group $\mathcal{G}$ associated with left module V. Let $\mathcal{H}$ be a subgroup of $\mathcal{G}$. Then $\mathcal{M}(\mathcal{H})$ is a projection matrix onto the fixed point subspace of $\mathcal{H}$ in V.*

Proof. We prove first that the range of $\mathcal{M}(\mathcal{H})$ is equal to the fixed point subspace of $\mathcal{H}$, and then that $\mathcal{M}(\mathcal{H})$ is a projection matrix.

1. First we show that the range of $\mathcal{M}(\mathcal{H})$ is a subspace of the fixed point subspace of $\mathcal{H}$. Let $w = \mathcal{M}(\mathcal{H})v$ for some $v \in V$, h be any element of $\mathcal{H}$. Now we have

$$M(h)w = M(h)(\mathcal{M}(\mathcal{H})v) = (M(h)\mathcal{M}(\mathcal{H}))v = \mathcal{M}(\mathcal{H})v = w.$$

Thus the range of $\mathcal{M}(\mathcal{H})$ is a subspace of the fixed point subspace of $\mathcal{H}$.

Now let v be any vector in the fixed point subspace of $\mathcal{H}$; i.e. $M(h)v = v \ \forall h \in \mathcal{H}$. Summing over $\mathcal{H}$ gives $\mathcal{M}(\mathcal{H})v = v$; i.e. v is in the range of $\mathcal{M}(\mathcal{H})$. Therefore the range of $\mathcal{M}(\mathcal{H})$ is equal to the fixed point subspace of $\mathcal{H}$.

2. Projection matrices are those which are both idempotent and self-adjoint. $\mathcal{M}(\mathcal{H})$ is idempotent:

$$\begin{aligned} \mathcal{M}(\mathcal{H})^2 &= \left(\frac{1}{|\mathcal{H}|} \sum_{g \in \mathcal{H}} M_g \right) \left(\frac{1}{|\mathcal{H}|} \sum_{g' \in \mathcal{H}} M_{g'} \right) \\ &= \left(\frac{1}{|\mathcal{H}|} \right)^2 |\mathcal{H}| \sum_{g'' \in \mathcal{H}} M_{g''} = \mathcal{M}(\mathcal{H}). \end{aligned}$$

$\mathcal{M}(\mathcal{H})$ is also self-adjoint (i.e. its transpose is equal to its complex conjugate), because it is real, and also symmetric as M is orthogonal. Hence $\mathcal{M}(\mathcal{H})$ is a projection matrix.

This completes the proof of the theorem. □

The above result is an extremely pleasing property of the characteristic matrix.

Corollary 3.5. *For orthogonal M, the trace of the characteristic matrix $\mathcal{M}(\mathcal{H})$ is equal to its rank, which is also equal to the dimension of the fixed point subspace of $\mathcal{H}$.*

Proof. The eigenvalues of any projection matrix are 0 and 1. The trace of a matrix is equal to the sum of the eigenvalues. The rank of a matrix is the number of non-zero eigenvalues. The dimension of the fixed point subspace of $\mathcal{H}$ is the dimension of the range of $\mathcal{M}(\mathcal{H})$. Hence all three quantities are the same. □

It is helpful to view the parameter dimension as the dimension of the fixed point subspace of $\mathcal{H}$. It is often easy to see what this value is. In particular the parameter dimension will be n (the maximum possible) if and only if $\mathcal{H}$ is the trivial subgroup.

The choice of the subgroup $\mathcal{H}$ is evidently crucial to the structure and power of the resulting PRN. It therefore affects the discriminability and learning abilities of the network, as will be discussed briefly in Section 6.

3.4. A minimal parameter set

We have a formula (in Theorem 3.3) expressing the weights in terms of a set of n parameters. However we know that the parameter dimension is the rank of the subgroup characteristic matrix, which is *at most* n and certainly will not be equal to n in general. The question then arises as to how to express the weights of the network in terms of a minimal parameter set.

The answer to this problem again lies in the reduction to Echelon form, this time of the characteristic matrix. For if the matrices $E_1, E_2, \ldots, E_k$ define a sequence of operations which convert $\mathcal{M}(\mathcal{H})$ to row-reduced Echelon form, then

$$\rho\mathcal{M}(\mathcal{H}) = \overline{\rho}\overline{\mathcal{M}(\mathcal{H})},$$

where $\overline{\mathcal{M}(\mathcal{H})} = E_k \ldots E_2 E_1 \mathcal{M}(\mathcal{H})$ is the Echelon form of $\mathcal{M}(\mathcal{H})$ and $\overline{\rho} = \rho E_1^{-1} E_2^{-1} \ldots E_k^{-1}$ is a new parameter vector. We can therefore replace ρ by $\overline{\rho}$ and $\mathcal{M}(\mathcal{H})$ by $\overline{\mathcal{M}(\mathcal{H})}$ in the formula for the weight matrix given in Theorem 3.3. $\overline{\mathcal{M}(\mathcal{H})}$ will contain a number of non-zero rows equal to $p(M,P)$, and hence only the corresponding parameters in $\overline{\rho}$ will be used in the formula for W.

Standard algorithms exist for reduction of a matrix to Echelon form. Thus the entire process of calculation of a weight matrix in terms of a minimal set of parameters can easily be automated.

4. Other representations

We now consider the further categories of group representations arising from Theorem 2.4.

The second category of group representations which are preserved by a broad class of activation functions is the inversion representations. We therefore deal with these next.

4.1. Inversion representations

Recall an inversion matrix is a diagonal matrix with diagonal entries of ± 1. An *inversion representation* is a matrix representation in which all group representative matrices can be written as the product of an inversion matrix and a permutation matrix. In effect, then, each representative matrix in an inversion representation is a permutation matrix in which some of the 1's may have been replaced by -1's. Like permutation representations, inversion representations are always orthogonal.

We let Z denote an arbitrary inversion representation. For any $g \in \mathcal{G}$, we can decompose $Z(g)$ into an inversion matrix $\psi(g)$ and a permutation matrix $P(g)$:

$$Z(g) = \psi(g)P(g)$$

For convenience we write $Z = \psi P$. It is easy to see that P is also a representation of $\mathcal{G}$.

We are now able to provide a classification of inversion representations $Z = \psi P$ for which P is transitive. When P is not transitive, we can rewrite Z as the direct sum of inversion representations which do have transitive associated permutation representations. We can then consider each subrepresentation separately.

Theorem 4.1. *Let $Z = \psi P$ be an inversion representation with P transitive of dimension m. Let $S = \{-1, -2, \ldots, -m, 1, 2, \ldots, m\}$ denote a set on which Z acts by permutation in the natural manner.*

Define a subgroup $\mathcal{H}$ of $\mathcal{G}$:

$$\mathcal{H} = \{g \in \mathcal{G} | g(1) = \pm 1\}.$$

Then one of two cases holds:

1. *Z is induced from a non-trivial alternating representation of the subgroup $\mathcal{H}$, and S forms a single orbit under Z.*
2. *Z is equivalent to P and partitions S into two orbits, either i or $-i$ being in each orbit for all $i \in 1 \ldots m$.*

Proof. We give only a sketch of the proof here. Since P is transitive, either i or $-i$ must be in the same orbit as 1 for all i, and the other must be in the same orbit as -1. The orbits of 1 and -1 may or may not be distinct, but they clearly cover S, and so there are either one or two orbits of S under Z.

1. Assume S is a single orbit under Z. Define the class function α on $\mathscr{H}$ by $\alpha(h) = Z(h)_{(1,1)} = \pm 1$ for all $h \in \mathscr{H}$.

It is easy to show that α is an alternating representation of $\mathscr{H}$; furthermore it is non-trivial since 1 and -1 are in the same orbit. Let $\mathrm{ind}_{\mathscr{H}}^{\mathscr{G}}\alpha$ denote the representation of $\mathscr{G}$ induced from α.

Let $h_1, h_2, \ldots, h_m$ denote the coset representatives of $\mathscr{H}$, ordered so that $h_i(1) = \pm i$ for all i.

By the definition of an induced representation we get

$$(\mathrm{ind}_{\mathscr{H}}^{\mathscr{G}}\alpha)(g)_{(i,j)} = Z(h_i g h_j^{-1})_{(1,1)} \quad \forall g \in \mathscr{G}.$$

It can be shown that $Z(h_i g h_j^{-1})_{(1,1)} = Z(g)_{(i,j)}$. Thus we get that $\mathrm{ind}_{\mathscr{H}}^{\mathscr{G}}\alpha = Z$, which is our required result.

2. Now assume S is partitioned into two orbits under Z. Define the matrix T by

$$T_{(i,j)} = \begin{cases} 0 & \text{if } i \neq j, \\ 1 & \text{if } i \text{ is in the same orbit as } 1, \\ -1 & \text{if } i \text{ is in the same orbit as } -1. \end{cases}$$

For any $g \in \mathscr{G}$ we now get

$$(TZ(g)T^{-1})_{(i,j)} = T_{(i,i)}\psi(g)_{(i,i)}P(g)_{(i,j)}T_{(j,j)} = P(g)_{(i,j)}.$$

This last step follows by consideration of the definition of T. Thus we get that $TZ(g)T^{-1} = P(g)$ for all $g \in \mathscr{G}$, i.e. P is equivalent to Z. □

When Z falls into the first category of inversion representations listed in the theorem, we refer to it as a *one-orbit* inversion representation. Otherwise we call Z a *two-orbit* inversion representation.

Having divided the inversion representations into two subclasses, we are now in a position to give formulae for the parameter dimension and weight matrix of a pair (M, Z), where M is an arbitrary orthogonal representation of $\mathscr{G}$.

Corollary 4.2. *Let $Z = \psi P$ be an inversion representation of group $\mathscr{G}$, with P transitive. Let M be an arbitrary orthogonal representation of $\mathscr{G}$. Define two subgroups $\mathscr{F}$, $\mathscr{H}$ of $\mathscr{G}$ by :*

$$\mathscr{H} = \{g \in \mathscr{G} \mid g(1) = \pm 1\}, \quad \mathscr{F} = \{g \in \mathscr{G} \mid g(1) = 1\}$$

according to the action of Z on the set $\{-1, -2, \ldots -m, 1, 2, \ldots m\}$. Let $h_1, h_2, \ldots h_m$ denote the right coset representatives of $\mathscr{H}$, ordered so that $h_i(1) = \pm i$ for all $i \in 1 \ldots m$. Let $f_1, f_2, \ldots f_{2m}$ denote the coset representatives of $\mathscr{F}$, ordered so that $f_i(1) = i$, $f_{m+i} = -i$ for $i \in 1 \ldots m$.

Then we have:

1. *If Z is a one-orbit representation, $p(M,Z)$ is* $\mathrm{Trace}(\mathscr{M}(\mathscr{F})) - \mathrm{Trace}(\mathscr{M}(\mathscr{H}))$, *and*

$$W_{M,Z} = \begin{pmatrix} \rho\mathscr{M}(\mathscr{F})(M(f_1) - M(f_{m+1})) \\ \rho\mathscr{M}(\mathscr{F})(M(f_2) - M(f_{m+2})) \\ \vdots \\ \rho\mathscr{M}(\mathscr{F})(M(f_m) - M(f_{2m})) \end{pmatrix}.$$

2. *If Z is a two-orbit representation, $p(M,Z)$ is the trace of the characteristic matrix $\mathscr{M}(\mathscr{H})$ of $\mathscr{H}$ in the representation M, and the weight matrix is given by:*

$$W_{M,Z} = \begin{pmatrix} \eta_1\rho\mathscr{M}(\mathscr{H})M(h_1) \\ \eta_2\rho\mathscr{M}(\mathscr{H})M(h_2) \\ \vdots \\ \eta_m\rho\mathscr{M}(\mathscr{H})M(h_m) \end{pmatrix},$$

where $\eta_i = \pm 1$; $\eta_i = 1$ iff i is in the same orbit as 1,
where ρ is a vector of independent parameters.

Proof. Again we give only a sketch proof, considering each of the two cases in turn.

1. Suppose Z is a one-orbit representation. By Theorem 4.1, $Z = \mathrm{ind}_{\mathscr{H}}^{\mathscr{G}}\alpha$ for a non-trivial alternating representation α of $\mathscr{H}$. Let $\mathrm{res}_{\mathscr{H}}^{\mathscr{G}}A$ denote the restriction of representation A of $\mathscr{G}$ to the subgroup $\mathscr{H}$. For any representation A, let $\chi(A)$ denote its character. By Lemma 2.7 and Frobenius' Reciprocity Theorem we have

$$\begin{aligned} p(M,Z) &= \langle\chi(M),\chi(Z)\rangle_{\mathscr{G}} \\ &= \langle\chi(M),\chi(\mathrm{ind}_{\mathscr{H}}^{\mathscr{G}}\alpha)\rangle_{\mathscr{G}} \\ &= \langle\chi(\mathrm{res}_{\mathscr{H}}^{G}M),\alpha\rangle_{\mathscr{H}} \\ &= \frac{1}{|\mathscr{H}|}\sum_{h\in\mathscr{H}}\alpha(h).Tr(M(h)) \\ \ldots &= \mathrm{Trace}(\mathscr{M}(\mathscr{F})) - \mathrm{Trace}(\mathscr{M}(\mathscr{H})). \end{aligned}$$

Now let P' denote the permutation representation corresponding to subgroup $\mathscr{F}$ of $\mathscr{G}$, i.e. F is the stabilizer of a point under P'. It can be shown that

$$P'(g) = \frac{1}{2}\begin{pmatrix} P(g)+Z(g) & P(g)-Z(g) \\ P(g)-Z(g) & P(g)+Z(g) \end{pmatrix}$$

Defining $D = ((1-1)\otimes I_m)$, where $\otimes$ denotes direct product and I_m is the identity matrix of degree m, we find $DP'(g) = Z(g)D$ for all $g \in \mathscr{G}$. Using the formula in Lemma 2.5, we can show that $W_{M,Z} = DW_{M,P'}$. By Theorem 3.3 we obtain the required result.

2. Suppose Z is a two-orbit representation. By Theorem 4.1, Z is equivalent to P. By Lemma 2.7, $p(A,B) = p(A,C)$ whenever B and C are equivalent representations.

Hence $p(M,Z) = \text{Trace}(\mathscr{M}(\mathscr{H}))$. Furthermore, by the proof of Theorem 4.1 we see that $TZT^{-1} = P$, where T is as defined in the earlier proof. We also have

$$W_{A,P}A(g) = TZ(g)T^{-1}W_{A,P} \quad \forall g \in \mathscr{G}$$
$$\Leftrightarrow \; TW_{A,P}A(g) = Z(g)TW_{A,P} \quad \forall g \in \mathscr{G}.$$

Hence $TW_{A,P} = W_{A,Z}$. Theorem 3.3 gives a formula for $W_{A,P}$, by the definition of T we reach the required result. □

We have now classified inversion representations and given formulae for the weight matrices in GRNs involving these representations.

4.2. Miscellaneous representations

By Theorem 2.4 any group representation other than an inversion or permutation, when used in an non-input layer of a GRN, places severe constraints on the activation function of the nodes of that layer.

Let A denote an arbitrary representation of $\mathscr{G}$ which is not a permutation, inversion or unit-row representation. If any matrix of A contains a negative entry, Theorem 2.4 tells us that the activation function in a GRN layer associated with A must be linear.

Linear functions are of no use as activation functions in a feedforward network, since they have no processing power. A hidden layer of neurons, each with the same linear activation function, is redundant and can be removed by combining connections leading into and out of this layer. Thus such a representation A is of no use in a GRN.

Now suppose every representative matrix of A contains only positive entries. By Theorem 2.4 this will allow the use of semilinear activation functions. However, such representations would seem to be uncommon. A trivial example is the two-dimensional representation of $S_2 = \{e, g\}$ given by

$$A(e) = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \qquad A(g) = \begin{pmatrix} 0 & 2 \\ \frac{1}{2} & 0 \end{pmatrix}.$$

The final category of representations is that of unit-row representations. If these are used, however, we are restricted to the use of affine activation functions, which also have no real processing power.

5. Example GRN

We now present a simple example of a Group Representation Network, just to illustrate the theory above. The invariance group will be the symmetric group of degree 3:

$$\mathscr{G} = S_3 = \{e, g_1, g_1^2, g_2, g_1g_2, g_1^2g_2\}.$$

We will build a four-layer network. The representations corresponding to each layer are as follows.

1. The input layer corresponds to the two-dimensional irreducible representation A_0, which is the representation obtained by regarding g_1 as a 120° rotation about the origin and g_2 as a reflection in the y-axis, action being on the Euclidean plane.

$$A_0(g_1) = \begin{pmatrix} -\frac{1}{2} & -\frac{\sqrt{3}}{2} \\ \frac{\sqrt{3}}{2} & -\frac{1}{2} \end{pmatrix}, \qquad A_0(g_2) = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}.$$

2. The first hidden layer corresponds to an inversion representation A_1 induced from the alternating representation of the subgroup $\{e, g_2\}$:

$$A_1(g_1) = \begin{pmatrix} 0 & 0 & -1 \\ -1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}, \qquad A_1(g_2) = \begin{pmatrix} -1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & -1 & 0 \end{pmatrix}.$$

3. The second hidden layer corresponds to the regular representation A_2.

$$A_2(g_1) = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}, \qquad A_2(g_2) = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

4. The output layer corresponds to the trivial representation A_3 of $\mathscr{G}$.

Thus the numbers of nodes in each layer are 2, 3, 6 and 1 respectively. We take each layer to have connections leading only to the layer above. The activation functions involved are arbitrary, save that those in the first hidden layer must be odd.

By applying Theorem 3.3 and Corollary 4.2 we find the number of parameters in each layer:

$$p(A_0, A_1) = 1, \quad p(A_1, A_2) = 3, \quad p(A_2, A_3) = 1.$$

We can also calculate the generalized weight matrices :

$$W_{A_0,A_1} = \begin{pmatrix} 2\alpha_1 & 0 \\ \alpha_1 & -\sqrt{3}\alpha_1 \\ \alpha_1 & \sqrt{3}\alpha_1 \end{pmatrix}, \quad W_{A_1,A_2} = \begin{pmatrix} \alpha_2 & \alpha_3 & \alpha_4 \\ -\alpha_3 & \alpha_4 & -\alpha_2 \\ -\alpha_4 & -\alpha_2 & \alpha_3 \\ -\alpha_2 & -\alpha_4 & -\alpha_3 \\ \alpha_4 & -\alpha_3 & \alpha_2 \\ \alpha_3 & \alpha_2 & -\alpha_4 \end{pmatrix}, \quad W_{A_2,A_3} = \begin{pmatrix} \alpha_5 \\ \alpha_5 \\ \alpha_5 \\ \alpha_5 \\ \alpha_5 \\ \alpha_5 \end{pmatrix}^{\mathrm{T}}.$$

Here the free parameters of the network are $\{\alpha_1, \ldots, \alpha_5\}$. Additional degrees of freedom can also be incorporated, for example by the addition of a variable 'threshold' to the activation functions in the second hidden layer and the output layer.

The GRN specified above will be invariant under the action of A_0 on the input layer.

6. Practical issues

As with Symmetry Networks, Group Representation Networks may have a problem with discriminability. We say that a network *discriminates perfectly* over a group $\mathcal{G}$ if any two inputs which are in distinct orbits under $\mathcal{G}$ can also be distinguished by the network. Discriminability is a property of a particular network structure, rather than one of the invariance problem itself. Intuitively, we expect perfect discriminability to be more common in networks with a greater number of free parameters. Experimental observations have supported this conjecture.

Consider a problem which can be specified by the classification of a finite number of orbits under group $\mathcal{G}$ (those inputs not in these orbits implicitly receiving a distinct classification). A perfectly discriminating GRN can perform this pattern classification without the need of training, which clearly saves a lot of computational time. This is done simply by comparing the output of the network for a new (test) pattern with the outputs of the prototype (training) patterns.

We also save computational space owing to the fact that the classification of an orbit is given by the classification of a single input in that orbit. A conventional feedforward network learning the same problem would require the classification of a number of inputs in the orbit (the interpolative abilities of the network hopefully extending the classification to the whole orbit).

In most practical problems, some degree of generalization will be required of the network. The network must then be trained on a well-chosen set of prototype patterns. Any standard learning algorithm (for example backpropagation) can be readily adapted to reflect the fact that it is the parameters of the network's weights that are adaptable, rather than the weights themselves.

We also expect a GRN to train faster and/or to generalize more successfully than a network without in-built invariance. This is simply because we are giving the network information a priori and therefore it inevitably knows more about the problem than a conventional network which would in effect have to *learn* the invariance in order to generalize equally well. In [14], the generalization properties of Symmetry Networks are investigated. Since GRNs are simply more general versions of Symmetry Networks, we expect similar properties to hold for them.

6.1. Simulations

In [13], a number of experiments were carried out using Symmetry Networks on the graph isomorphism problem. In these experiments the Symmetry Networks trained significantly faster than networks without in-built invariance.

Since the development of the theory in this paper, we have performed two further sets of experiments comparing the performance of GRNs with that of conventional feedforward networks. Firstly, we tested a class of GRNs on the parity problem, where the classification of a string is invariant under the inversion of an even number of bits. The inputs were ± 1-valued; hence such an inversion is a linear transformation of the input. We were able to present the problem to an invariant net by just two input samples (one positive, one negative); the invariance specified the remaining classifications. Simulations showed that the invariant networks trained significantly faster than conventional networks; the learning algorithm however sometimes failed to converge with the in-built constraints of the GRNs.

A second experiment was carried out on a simple character recognition problem, with invariance under 30° rotations and a vertical reflection. Seventy-two letters (with classifications A, M and X) were presented as training data, each specified by the coordinates of its key points. In this experiment, the invariant network was slower to train than the conventional network. However it correctly classified new inputs 100% successfully, whereas the conventional network achieved only 92% correct classification.

7. Conclusions

We have introduced the concept of Group Representation Networks. These are feedforward neural networks which are intrinsically invariant under the action on the input of an arbitrary finite linear group. GRNs include as a subclass Symmetry Networks, which have previously been used to achieve permutation invariance.

The structure of a GRN is determined by a choice of representation for each layer of the network. We have presented a classification of group representations according to their usefulness in these new networks. We have analyzed the local structure of GRNs employing the more applicable classes of representation. In particular we have presented formulae for the number of free parameters and the weight matrix of a given layer of connections in a GRN.

Appendix A: Proof of Theorem 2.4

For convenience we restate the theorem and its prerequisite definitions. For any real number a, we define the following functions:

$$l_a^+, l_a^- : \mathbb{R} \mapsto \mathbb{R}, \text{by } l_a^+(x) = \begin{cases} ax, & x \geqslant 0, \\ \text{undefined}, & x < 0, \end{cases} \quad l_a^-(x) = \begin{cases} \text{undefined}, & x > 0, \\ ax, & x \leqslant 0. \end{cases}$$

$\Phi = \{\phi : \mathbb{R} \mapsto \mathbb{R} | \phi$ intersects l_a^+ at an infinite number of places for at most one value of a, and ϕ intersects l_a^- at an infinite number of places for at most one value of $a\}$

Theorem 2.4. *Let A be a finite-dimensional representation of the group $\mathcal{G}$ acting on a real vector space, $f : \mathbb{R} \mapsto \mathbb{R}$ a function which is such that $f_0 : \mathbb{R} \mapsto \mathbb{R}$ defined by $f_0(x) = f(x) - f(0)$ is in Φ. Then f preserves A if and only if one of the following conditions holds :*

1. *A is a permutation representation.*
2. *A is an inversion representation and f is odd.*
3. *A is a unit-row representation (i.e. every row of every matrix of A sums to 1) and f is affine.*
4. *A is a positive representation (i.e. every entry of every matrix of A is non-negative) and f is semilinear (i.e. there exist reals k_1 and k_2 such that $f(x) = k_1 x$ for $x \geqslant 0$, $f(x) = k_2 x$ for $x < 0$).*
5. *f is linear.*

Before starting on the main body of the proof we present a number of preliminary lemmas. Throughout, A denotes a finite-dimensional representation of the group $\mathcal{G}$ acting on a real vector space, and n denotes the dimension of A.

Lemma A.1. *If M is an invertible matrix which cannot be written as the product of inversion and permutation matrices, then either M contains an entry other than $-1, 0, 1$ or else M contains a row with at least two non-zero elements.*

Proof. Let M denote an invertible matrix which contains no row with more than one non-zero element and no entries other than $-1, 0$ and 1. No row of M can be all-zero, and no two rows of M can be linearly dependent. Thus every row of M must contain exactly one non-zero entry and similarly for every column. We can therefore write $M = M_1 M_2$, where M_1 is diagonal with entries of 1 in rows where M has entries of 1 and entries of -1 in other rows. M_2 will have zero entries in the same places as M and 1 entries elsewhere. Thus M_1 is an inversion matrix and M_2 a permutation matrix. □

Lemma A.2. *If A is a positive representation, then any matrix M of A must have exactly one (positive) entry in each row and column.*

Proof. Let the inverse of M be denoted by N; N is in the representation A and hence must itself have only positive entries. Considering the entries of the product $MN = I$, we have

$$\forall i, j \text{ with } i \neq j \qquad \sum_{k=1}^{n} M_{ik} N_{kj} = 0.$$

Now choose an arbitrary value of i. The ith row of M must contain a non-zero element M_{ir} (or else it would not be invertible). This we know is positive, and so by the above formula we must have $N_{rj} = 0$ for all $j \neq i$. Since the rth row of N cannot be all zero, we must have $N_{ri} \neq 0$. But from this we conclude similarly that

$M_{jr} = 0$ for all $j \neq i$. Thus each column of M and each row of N contains exactly one non-zero entry. By reversing the roles of M and N in the argument above, we complete the proof. □

Corollary A.3. *A positive unit-row representation is a permutation representation.*

Lemma A.4. *The function $f : \mathbb{R} \mapsto \mathbb{R}$ preserves the representation A if and only if*

$$\forall i \in 1 \ldots n, v \in \mathbb{R}^n, g \in \mathcal{G} \quad f\left(\sum_{j=1}^{n} A(g)_{ij} v_j\right) = \sum_{j=1}^{n} A(g)_{ij} f(v_j). \tag{5}$$

Proof. This follows directly from the definition of preservation. □

Lemma A.5. *Let f be a function which preserves the representation A. Then either A is a unit-row representation or else f passes through the origin.*

Proof. Substituting $v = 0$ in Eq. (5), we obtain

$$\forall i \in 1 \ldots n, g \in \mathcal{G} \qquad f(0) = f(0) \sum_{j=1}^{n} A(g)_{ij}$$

Hence either $f(0) = 0$ or $\sum_{j=1}^{n} A(g)_{ij} = 1$ for all i and g. This is the required result. □

Lemma A.6. *Let f be a function which preserves A. Let τ be the sum of any number of entries in any row of a matrix of A. Then we have*

$$\forall x \in \mathbb{R} \qquad f(\tau x) = \tau f(x) + (1 - \tau) f(0) \tag{6}$$

Proof. Let e_i denote the ith column of the identity matrix. Suppose τ is the sum of entries in row r of matrix M of A. Define the subset S of $\{1, \ldots n\}$ by $\tau = \sum_{j \in S} M_{rj}$. Finally, define $v = \sum_{j \in S} x e_j$ for arbitrary x. Now from Eq. (5) with $i = r$ we have

$$f\left(\sum_{j \in S} M_{rj} x\right) = \sum_{j \in S} M_{rj} f(x) + \sum_{l \notin S} M_{rl} f(0)$$

$$\Rightarrow \quad f(\tau x) = \tau f(x) + \left(\left(\sum_{j=1}^{n} M_{rj}\right) - \tau\right) f(0).$$

Now by Lemma A.5, either $f(0) = 0$ or $\sum_{j=1}^{n} M_{rj} = 1$. Hence we obtain the required result. □

Lemma A.7. *Given any function f, the set of all real numbers τ which satisfy Eq. (6) (excluding $\tau = 0$) form a multiplicative group.*

Proof. Let T denote the set of solutions τ to equation (6). The associativity axiom follows trivially, and the identity axiom can be verified by substituting $\tau = 1$.

Let τ_1, τ_2 be in T. Now for any $x \in \mathbb{R}$ we have

$$\begin{aligned} f(\tau_1\tau_2 x) &= \tau_1 f(\tau_2 x) + (1 - \tau_1) f(0) \\ &= \tau_1 [\tau_2 f(x) + (1 - \tau_2) f(0)] + (1 - \tau_1) f(0) \\ &= \tau_1\tau_2 f(x) + (1 - \tau_1\tau_2) f(0). \end{aligned}$$

Hence Eq. (6) is satisfied for $\tau = \tau_1\tau_2$, and so T is closed under multiplication.

Finally, the proof that T is closed under inversion follows by rearrangement of Eq (6). □

Lemma A.8. *Let $\phi : \mathbb{R} \mapsto \mathbb{R}$ be a function in the set Φ. Suppose there exists a value of τ not in $\{-1, 0, 1\}$ such that*

$$\forall x \in \mathbb{R} \qquad \phi(\tau x) = \tau\phi(x). \tag{7}$$

Then ϕ is semilinear. Furthermore, if there exists such a τ which is negative, ϕ is linear.

Proof. Let a denote a real value for which l_a^+ intersects ϕ, and let x_1 denote a given point of intersection, i.e. $l_a^+(x_1) = \phi(x_1)$. Now we have

$$\phi(\tau^2 x_1) = \tau^2\phi(x_1) = \tau^2 l_a^+(x_1) = \tau^2 a x_1 = l_a^+(\tau^2 x_1)$$

(the last step since $\tau^2 x_1 > 0$). Hence τx_1 is another point of intersection. Similarly, $\tau^{2z} x_1$ is a point of intersectionfor all $z \in \mathbf{Z}$. Hence if ϕ intersects l_a^+ at all, it intersects it at infinitely many points.

From this, since ϕ is in Φ we conclude that ϕ intersects l_a^+ for at most one value of a, and clearly therefore for exactly one value, a_1 say. Therefore for positive x we have $\phi(x) = l_{a_1}^+(x)$. Similarly, we can deduce that for negative x and for some $a_2 \in \mathbb{R}$ we have $\phi(x) = l_{a_2}^-(x)$. Finally, for $x = 0$ it is clear that $\phi(x) = 0$. We have effectively deduced that ϕ is semilinear.

For the second part of the proof, we assume that τ is negative ($\tau \neq -1$). From the equation for a semilinear function we obtain expressions for $\phi(\tau x)$ and $\tau\phi(x)$. Equating these, we see that $a_1 = a_2$, i.e. f is linear. □

We now begin the main body of the proof of Theorem 2.4.

1. Let A be a permutation representation. Then each row of $A(g)$ contains exactly one non-zero entry, which is 1. Thus Eq (5) holds trivially for every i and any f.

2. Let A be an inversion representation. Hence any group representative matrix in A is the product of an inversion matrix and a permutation matrix. Any f preserves all permutation matrices; hence f preserves A if and only if it preserves the inversion matrices forming the matrices of A.

Therefore let M be such an inversion matrix, $M \neq I$. M must contain an entry of -1, and hence by Lemma A.6 we have

$$\forall x \in \mathbb{R} \qquad f(-x) = -f(x) + 2f(0).$$

However, since A is clearly not a unit-row representation, by Lemma A.5 we know $f(0) = 0$, and hence we have that f is odd. The converse, that any odd function preserves any inversion matrix, is easy to prove.

3. Let A be a representation which is neither an inversion nor a permutation representation. We will consider the remaining cases simultaneously. We assume A contains a matrix M which is neither an inversion nor a permutation (nor a product of the two). By Lemma A.1, there are two cases to consider regarding the entries of M:

(a) Assume M contains an entry M_{rs} not in $\{-1, 0, 1\}$. By Lemma A.6, Eq. (6) holds for $\tau = M_{rs}$.

(b) Conversely, assume M contains only entries of $-1, 0, 1$. Hence it contains a row with at least two entries of ± 1. Choose two of these entries, M_{rs} and M_{rt}, if possible such that they are both 1. We now have again two possibilities:

(i) $M_{rs} = M_{rt} = 1$. In this case, we can apply Lemma A.6 and obtain Eq. (6) with $\tau = M_{rs} + M_{rt} = 2$.

(ii) At least one of the entries M_{rs} and M_{rt} is -1. However, now A is certainly not a unit-row representation, since if it were, any entry of -1 in a row of M would necessitate at least two entries of $+1$ in that row, which would be a contradiction. Therefore by Lemma A.5, $f(0) = 0$, and by Lemma A.6 we have Eq. (6) for $\tau = -1$, i.e. f is odd. Hence we can say $f(M_{rs}x) = M_{rs}f(x)$, $f(M_{rt}x) = M_{rt}f(x)$.

Now we substitute $v = M_{rs}xe_s + M_{rt}xe_t$ (for arbitrary x) and $i = r$ in Eq. (5) and obtain

$$f(M_{rs}^2 x + M_{rt}^2 x) = M_{rs}f(M_{rs}x) + M_{rt}f(M_{rt}x) + f(0) \sum_{j=0, j\neq s,t}^{n} M_{rj}$$

$$\Rightarrow \quad f(2x) = 2f(x) + f(0) \sum_{j=0, j\neq s,t}^{n} M_{rj}$$

$$= 2f(x) + (1-2)f(0) \qquad (\text{since } f(0) = 0).$$

Thus, in either case, Eq. (6) holds for $\tau = 2$.

At this stage, we have found a value k not in $\{-1, 0, 1\}$ such that Eq. (6) holds for $\tau = k$. Recall that the function f_0 is defined by $f_0(x) = f(x) - f(0)$. From this we derive the following:

$$\forall x \in \mathbb{R} \qquad f_0(kx) = kf_0(x). \tag{8}$$

Hence we can apply Lemma A.8 for $\phi = f_0 \in \Phi$, and so f_0 must be a semilinear function. We now have the following two possibilities:

(a) A is a positive representation. By Corollary A.3, A cannot be a unit-row representation and thus, by Lemma A.5, $f(0) = 0$, i.e. $f_0 = f$ and f is semilinear.

We must also show that any semilinear function preserves any positive representation. This follows from Lemma A.2; we can see that a sufficient condition for f to preserve A is that $f(\alpha x) = \alpha f(x)$ for all $x \in \mathbb{R}$, $\alpha \in \mathbb{R}$, $\alpha > 0$, which is equivalent to semilinearity.

(b) A is not a positive representation. Then either k itself is negative and not equal to -1, or else k is positive and not equal to 1. In the latter case, let l denote a negative element of some matrix of A. By Lemma A.6, we have Eq (6) for $\tau = l$, and hence, by Lemma A.7, this equation also holds for $\tau = kl$. Both l and kl are negative, and at least one is not equal to -1.

So we have a negative value of τ, other than -1, for which Eq. (6), and hence Eq. (8), holds. By Lemma A.8, f_0 must be linear.

If A is a unit-row representation, we have effectively deduced that f must be affine. On the other hand, if A is not a unit-row representation, we have $f_0 = f$ and thus f is linear.

It is evident that any linear function preserves any representation, hence the only remaining part of the proof is to show that any affine function preserves any unit-row representation. This is not hard, and we omit it for brevity.

This concludes the proof.

References

[1] M. Clausen and U. Baum, Fast Fourier Transforms (Wissenschaftsverlag, Mannheim 1993).

[2] P.M. Cohn, Algebra, Vol. 2 2nd ed., (Wiley, Chicester, 1989).

[3] L. Fausett, Fundamentals of Neural Networks (Prentice-Hall, Englewood Cliffs, NJ, 1994).

[4] W. Fulton and J. Harris, Representation Theory (A First Course) (Springer, New York, 1991).

[5] J. Hertz, A. Krogh and R. Palmer, Introduction to the Theory of Neural Computation (Addison-Wesley, Redwood City, CA, 1991).

[6] Y. Li, Reforming the theory of invariant moments for pattern recognition, Pattern Recognition 25 (7) (1992) 723–730.

[7] W. McCulloch and W. Pitts, A logical calculus of the ideas immanent in nervous activity, Bull. Math. Biophys. 5 (1943) 115–133.

[8] M. Minsky and S. Papert, Perceptrons (MIT Press, Cambridge, 1969).

[9] C.W. Norman, Undergraduate Algebra (A First Course) (Clarendon Press, Oxford, 1986).

[10] S. Perantonis and P. Lisboa, Translation, rotation and scale invariant pattern recognition by high-order neural networks and moment classifiers, IEEE Trans. Neural Networks 3 (2) (1992) 241–251.

[11] F. Rosenblatt, Principles of Neurodynamics (Spartan, New York, 1962).

[12] D. Rumelhart, G. Hinton and R. Williams, Learning representations by back-propagating errors, Nature 323 (1986).

[13] J. Shawe-Taylor, Building symmetries into feedforward networks, Proc. of First IEE Conference on Artificial Neural Networks (1989) 158–162.

[14] J. Shawe-Taylor, Threshold network learning in the presence of equivalences, in: Proceedings of NIPS-4 (Morgan Kaufmann, San Mateo, 1992).

[15] J. Shawe-Taylor,Symmetries and discriminability in feedforward network architectures, IEEE Trans. Neural Networks 4 (5) (1993) 816–826.

[16] J. Shawe-Taylor and D. Cohen, The linear programming algorithm, Neural Networks 3 (1990) 575–582.

[17] L. Spirkovska and M. Reid, Robust position, scale and rotation invariant object recognition using higher-order neural networks, Pattern Recognition 25 (1992) 975–985.

[18] J. Wood and J. Shawe-Taylor, Theory of symmetry network structure, Technical Report, Department of Computer Science, Royal Holloway, University of London (1993).
[19] C. Yuceer and K. Oflazer, A rotation, scaling and translation invariant pattern classification system, Pattern Recognition 26 (5) (1993) 687–710.