ANALYSIS OF THE FIRST LAYER IN WEIGHTLESS NEURAL NETWORKS FOR 3_DIMENSIONAL PATTERN RECOGNITION

A. Vázquez-Nava,*
Escuela de Ingeniería. CENTRO UNIVERSITARIO MEXICO. DIVISION DE ESTUDIOS SUPERIORES
J. Figueroa Nazuno & E. Vargas Medina
Laboratorio de Sistemas Complejos
UNIVERSIDAD DEL VALLE DE MEXICO - LOMAS VERDES
Av. Paseo de las aves #1. Fracc. Lomas Verdes
Naucalpan Edo. de México MEXICO
FAX: (915)-2-38-73-61

ABSTRACT
The behavior of the first layer of a weightless artificial neural network is analyzed in this paper. The way in which the neural network receives external information changes according to different probability distribution functions that control data sampling from many different patterns. This paper describes the architecture of this system and shows the effect of the different probability distribution functions over 3-dimensional pattern recognition.

1. INTRODUCTION
Considering contemporary research in neural networks, it is known that combining a certain paradigm and hidden layers of neural networks gives as a result a powerful tool for pattern recognition and classification. However, there has been little development and interest in studying the first or receiving layer of artificial neural networks. Implicitly, it has been assumed that the first layer has an homogeneous (and almost an irrelevant) behavior which is associated to the global performance of the neural networks. This fact is against a large amount of physiological evidence indicating that the type and organization of receivers of biological neural networks have a great importance in their physiological behavior.

It is known that specific phenomena of full generality are associated to the behavior of the first layer (or receiving layer) of neural networks; the most known case is lateral inhibition. Earlier works have experimentally demonstrated the effects on pattern recognition and classification when the way in which the first layer of an artificial neural network receives external information is changed 4,5,6.

This paper describes a system composed by a fixed weightless neural network (Aleksander’s model)1,2,5. This system is capable of changing the way in which it receives external information. Patterns to be classified are sampled in many ways through the use of different probability distribution functions. These probability distribution functions determine the way in which the first layer is activated. The effect of the different probability distribution functions over 3-dimensional pattern recognition is studied in this paper.

2. PROCEDURE
The following are the necessary elements of a generalized weightless Neural Network (based on Aleksander’s Model) for 2D (2-dimensional) and 3D (3-dimensional) pattern learning and recognition. These elements are: Pattern representation array, Control arrays and probability distribution functions for data sampling, Mapping function, and Learning/Recognition array.

(1) Departamento de Sociología. UNIVERSIDAD AUTONOMA METROPOLITANA-IZTAPALAPA

549
2.1 Pattern Representation Array:
This array stores the color code of the points that belong to the learned/recognized pattern. In the case of 2D pattern learning/recognition, this array is represented by:

\[ V[i,j] := c \quad (1) \]

where "i", "j" are integers representing the coordinates of a point "P(i,j)" and "c" represents the color code for point "P". Generalizing for 3D, this array is represented by using a set of 2D arrays. This set is also an array:

\[ W[i,j,z] := c \quad (2) \]

this array is made of "p" matrices of the same type as "V":

\[ W[i,j,z] := V_z[i,j] := c \quad (z = 1, 2, 3, ..., p) \quad (3) \]

2.2 Control Matrixes and probability distribution functions for data sampling:
These matrices are required to perform the pattern learning and pattern recognition phases. The points of a pattern are sampled by using the values stored in the Control Matrixes. Different probability distribution functions ("P(X)"") compute these values that determine the way in which the first layer of the neural network is activated. The effect of the different probability distribution functions over 2D & 3D pattern recognition is studied in the next section.

For 2D pattern learning/recognition, 2 matrixes ("MI" & "MJ") store the coordinates of the points to be sampled. Matrix "MI" stores the values of coordinates "i" and matrix "MJ" stores the values of coordinates "j" of sampled elements from the Representation array "V". For 3D, there are 2 Control Matrixes ("MI_z" & "MJ_z") that are associated to each array "V_z" belonging to set "W". The following expressions compute the values stored by Control Matrixes:

\[ MI_z[i_{MI}, j_{MJ}] := INT((m + 1) * P(X)) \quad (4) \]
\[ MJ_z[i_{MI}, j_{MJ}] := INT((n + 1) * P(X)) \quad (5) \]

where "m" represents the number of rows of "V" and "n" represents the number of columns of the same array. "P(X)" is a probability distribution function with "X" as a random variable and 0 <= P(X) <= 1.

The dimension of matrixes "MI" and "MJ" is computed as a function of the dimension of matrix "V". Given "V" which capacity is "m*n", then:

\[ m_{MI} = m_{MJ} := m * 2 \quad (6) \]
\[ n_{MI} = n_{MJ} := n / 2 \quad (7) \]

where "m_{MI}" is the number of rows belonging to "MI" and "MJ"; "n_{MI}" is the number of columns belonging to these matrixes.

2.3 Mapping Function:
The main purpose of Control Matrixes is to control the sampling of elements belonging to matrix "V" (for 2D pattern learning and recognition) or matrixes "V_z" (for 3D). These sampled elements are the inputs for computing a Mapping Function "f". The Mapping Function "f" maps the sampled elements of "V" to elements belonging to the Learning/Recognition array. This Mapping Function computes the address of the "at\(h\)" element of the Learning/Recognition array. This mapping is performed with values "k", "l" and matrixes "MI", "MJ" & "V" as inputs. The Mapping Function has the following structure:

\[ Address := f(V[MI[k,l], MJ[k,l], max cod, k,l]) \quad (8) \]

For 2D pattern learning and recognition, the Mapping Function "f" is proposed as:
\[
\alpha := 1 + (k - 1) * \text{max cod}^{n_{\text{m}}} + \sum_{l=0}^{n_{\text{m}}-1} (V[M_l[k,l], M_J[k,l]] * \text{max cod}^l) \quad (9)
\]

where "k", "l" are the coordinates of the elements of matrixes "MI" and "MJ". "maxcod" is the maximum number of color codes used for pattern representation. "V[M_l[k,l], M_J[k,l]]" is an element which is sampled from "V", using the specified coordinates in matrixes "MI" & "MJ".

For 3D pattern learning and recognition, the Mapping Function "f_2" is proposed as:

\[
\alpha_z := 1 + (k - 1) * \text{max cod}^{n_{\text{m}}} + \sum_{l=0}^{n_{\text{m}}-1} (V_z[M_l[k,l], M_J[z][k,l]] * \text{max cod}^l) \quad (10)
\]

where "k","l" are coordinates of elements belonging to matrixes "MI" & "MJ". "maxcod" represents the maximum number of color codes for pattern representation. "V_z[M_l[k,l], M_J[z][k,l]]" is a sampled element from the "zth" matrix "V" which belongs to set "W".

2.4 Learning/Recognition Array and Learning Phase:

During the learning phase, the Mapping Function "f" is computed in such a way that the contents of the mapped elements of the Learning/Recognition Array (array "A") will be updated. These elements are referenced by the addresses which are computed by the Mapping Function "f". For 2D, each element of this array is composed by a set of bits, each one is related to a pattern class. If the "kth" bit of an element is set "on", then this fact implies that this element belongs to the pattern class "k". A single element of "A" may have several bits set "on" at different positions. This situation means that a single element of "A" may belong to different pattern classes.

For 3D, the same approach is followed. However, the Learning/Recognition array is composed by columns, each column represents a single array "A". Therefore, for 3D pattern learning and recognition, the Learning/Recognition array "B" is a set of arrays of type "A". The values "a_z", which are computed by the mapping function, represent the addresses of mapped elements belonging to the "zth" column of "B" (see fig. 1).

2.5 Mechanism for pattern learning:

The pattern to be learned by the Weightless Neural Network, is stored in the Pattern Representation Array ("V" for 2D, "W" for 3D). The coordinates of sampled elements belonging to the pattern are computed using a probability distribution function "P(X)". These coordinates are stored in the Control Matrixes for data sampling ("MI" & "MJ" for 2D, "MI_z" & "MJ_z" for 3D). These matrixes control the sampling of elements from the Pattern Representation Array. The sampled elements are used as inputs for the Mapping Function ("f" for 2D, "f_2" for 3D), so the addresses of elements belonging to the Learning/Recognition Array are computed. The contents of these mapped elements are updated by setting "on" the associated bit to the class "s". The learned pattern belongs to this class "s". In case of 2D pattern learning, "n_{MI}" elements of the Learning/Recognition Array will be updated (see figs. 1 & 2). For 2D, this update is performed by using the following expression:

\[
A[a] := A[a] \text{OR} s \quad (11)
\]

where "s" represents the class to which the learned pattern belongs, and "a" is the address computed by the Mapping Function "f".

In the case of 3D pattern learning, "p * n_{MI}" elements of the Learning/Recognition Array will be updated. For 3D, this update is performed by:

\[
B[a_z, z] := B[a_z, z] \text{OR} s \quad (12)
\]

where "s" represents the class to which the learned pattern belongs, and "a_z" represents the address of the
element to be updated in the \( z \)th column of array \( B \).

### 2.6 Mechanism for pattern recognition:

The pattern, which is to be classified by the neural network, is stored in the Pattern Representation Array ("V" for 2D, "W" for 3D). Using the same Control Matrices for data sampling ("MI" & "M\( \_I \)" for 2D, "MI\( _Z \)" & "M\( \_I \)_\( _Z \)" for 3D), which were computed during the learning phase, the elements of the pattern are selected to be inputs for the Mapping Function (see fig. 2).

For each mapped element belonging to the Learning/Recognition Array, which address is computed by the Mapping Function, its content is analyzed. This analysis consists of checking the bits which are set "on" in each mapped element. The number of bits of a given class is counted. For 2D, the following expression counts the number of bits set "on" for each class:

\[
\text{count}[s] = \sum_{r=1}^{n_{\text{MD}}} A[a_{r,s}] \text{AND} \frac{2^s}{2^t}
\]

where \( a \) is computed by using the Mapping Function \( f \). \( \text{maxclass} \) is the maximum number of pattern classes specified during learning phase.

For 3D, the following expression computes the count:

\[
\text{count}[s] = \sum_{z=1}^{p} \sum_{r=1}^{n_{\text{MD}}_z} B[a_{z,r,s}] \text{AND} \frac{2^s}{2^t}
\]

where \( p \) is the number of matrices \( V_z \) contained in set \( W \), and \( B \) is the Learning/Recognition array. \( a_{z,r,s} \) is the address of the mapped element which is computed by the Mapping Function \( f_{z,s} \).

Once the count for each pattern class has been computed, the counts are compared among each other. The highest count for a given pattern class \( k \) implies that the weightless neural network recognizes the input pattern as belonging to class \( k \). Pattern recognition may be measured by a score which ranges from 0 to 1. Each score is computed as a function of each count. For 2D, the score related to each pattern class is computed by the following expression:

\[
\text{score}[s] = \frac{\text{count}[s]}{n_{\text{MD}}} \quad \text{for } s = 0, 1, ..., \text{maxclass} - 1
\]

For 3D, the following expression computes each score:

\[
\text{score}[s] = \frac{\text{count}[s]}{p \times n_{\text{MD}}} \quad \text{for } s = 0, 1, ..., \text{maxclass} - 1
\]

### 3. EXPERIMENTS AND RESULTS

A factorial experimental design \( 3 \times 3 \times 4 \) (3 learned classes, 3 training conditions and 4 probability distribution functions) was implemented for testing the described artificial neural network. Each pattern was drawn with 3 different colors and using a \( 16 \times 16 \times 5 \) Pattern Representation Array.

The percentage and mean score of properly recognized patterns, and also standard deviation were measured for each experimental condition. The results of this analysis are shown in table 1.

The contents of table 1 were used as inputs for a Three-way Analysis of Variance (ANOVA). The results of this analysis clearly show that pattern recognition (performed by the weightless artificial neural

552
network in this paper) is affected by the number of learned classes (F=6.717, p<0.0001), the number of training patterns (F=378.170, p<0.0001) and the Probability Distribution Functions used for data sampling (F=15.014, p<0.0001). This analysis has also shown highly significant effects over pattern recognition due to the number of training patterns that interacts with the type of probability distribution (F=6.716, p<0.001); minor significant effects are associated with interaction between number of learned classes and number of training patterns (F=2.188, p<0.05). These effects can clearly be seen in figs. 3,4,5.

The number of learned patterns (for each class) has a direct relationship with the performance associated with pattern recognition and with the mean score of recognized classes. As the number of learned patterns increases, the percentage and mean score of recognized classes also increases.

It also can be noted that the percentage and mean score are not affected (in a negative way) as the number of used classes during learning phases increases.

4. CONCLUSIONS AND DISCUSSION

The results, which are described above, show in a very clear manner that different types of data sampling (used for activating the first layer of neural networks) have important and systematic effects on the global behavior of neural networks. It has been assumed that the first layer has a passive and unimportant behavior associated to the performance of the neural networks. However, this and earlier works4,5,6 show the way in which experimental manipulation of the first layer affects and facilitates pattern recognition made by a neural network. This fact not only has experimental importance but also shows that these theories and formal analysis about neural networks should be reconsidered in a theoretical way.
Generating coordinates of elements to be sampled from Pattern representation arrays \( \text{"Vz"} \)

Storing pattern to be learned and belonging to a specific class \( \text"s" \)

Sampling elements from the 3D Pattern Representation Arrays

Calculating mapped elements into Learning/Recognition Array by using Mapping Function

Updating mapped elements

\[
B[a, z] := B[a, z] \text{ or } 2^s
\]

FIG. 1.
Mechanism for 3D pattern recognition in a weightless artificial neural network

1. Generating coordinates of elements to be sampled from Pattern representation arrays "Vz"

2. Storing pattern to be classified

3. Sampling elements from the 3D Pattern Representation Arrays

4. Calculating Hamming distance from each class, according to each address

5. Classifying input pattern as belonging to the class with the highest score

FIG. 2.
Effects over pattern recognition due to different Probability Distribution Functions under various experimental conditions.

FIG. 3

Effects over pattern recognition when 2 classes are learned

Number of patterns for neural network training

Symology:
- Normal Distribution
- Uniform Distribution
- Cauchy Distribution
- Beta Distribution

FIG. 4

Effects over pattern recognition when 3 classes are learned

FIG. 5

Effects over pattern recognition when 6 classes are learned

Number of patterns for neural network training

556
### Probability Distribution Functions

<table>
<thead>
<tr>
<th>Number of learned classes:</th>
<th>with training</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>65.000 %</td>
<td>80.000 %</td>
<td>55.000 %</td>
<td>80.000 %</td>
<td>n = 20</td>
</tr>
<tr>
<td></td>
<td>0.6336</td>
<td>0.3985</td>
<td>0.2870</td>
<td>0.3977</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.3300</td>
<td>0.1600</td>
<td>0.1749</td>
<td>0.1756</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>90.000 %</td>
<td>90.477 %</td>
<td>80.000 %</td>
<td>85.000 %</td>
<td>n = 20</td>
</tr>
<tr>
<td></td>
<td>0.8760</td>
<td>0.8436</td>
<td>0.8786</td>
<td>0.8386</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.2040</td>
<td>0.2060</td>
<td>0.1643</td>
<td>0.1849</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>100.000 %</td>
<td>100.000 %</td>
<td>73.300 %</td>
<td>83.300 %</td>
<td>n = 20</td>
</tr>
<tr>
<td></td>
<td>1.000</td>
<td>1.000</td>
<td>0.9808</td>
<td>0.9642</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.000</td>
<td>0.000</td>
<td>0.0434</td>
<td>0.0070</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>51.725 %</td>
<td>66.670 %</td>
<td>46.600 %</td>
<td>73.300 %</td>
<td>n = 20</td>
</tr>
<tr>
<td></td>
<td>0.1916</td>
<td>0.3293</td>
<td>0.4551</td>
<td>0.2869</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.2434</td>
<td>0.1768</td>
<td>0.3110</td>
<td>0.1749</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>87.500 %</td>
<td>84.375 %</td>
<td>86.600 %</td>
<td>80.000 %</td>
<td>n = 20</td>
</tr>
<tr>
<td></td>
<td>0.9097</td>
<td>0.8886</td>
<td>0.8542</td>
<td>0.8882</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.1705</td>
<td>0.1702</td>
<td>0.1922</td>
<td>0.1555</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>92.105 %</td>
<td>89.740 %</td>
<td>87.500 %</td>
<td>87.500 %</td>
<td>n = 20</td>
</tr>
<tr>
<td></td>
<td>0.9833</td>
<td>0.9735</td>
<td>0.9760</td>
<td>0.9602</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.0684</td>
<td>0.0703</td>
<td>0.0523</td>
<td>0.0922</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>57.143 %</td>
<td>60.000 %</td>
<td>61.660 %</td>
<td>78.330 %</td>
<td>n = 20</td>
</tr>
<tr>
<td></td>
<td>0.5585</td>
<td>0.5272</td>
<td>0.5701</td>
<td>0.4723</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.3241</td>
<td>0.2840</td>
<td>0.2907</td>
<td>0.2484</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>80.000 %</td>
<td>80.000 %</td>
<td>78.570 %</td>
<td>81.420 %</td>
<td>n = 20</td>
</tr>
<tr>
<td></td>
<td>0.8677</td>
<td>0.8693</td>
<td>0.8941</td>
<td>0.8854</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.2180</td>
<td>0.1839</td>
<td>0.1574</td>
<td>0.1687</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.7413</td>
<td>0.9531</td>
<td>0.9898</td>
<td>0.9875</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.1010</td>
<td>0.0214</td>
<td>0.0426</td>
<td>0.0830</td>
<td></td>
</tr>
</tbody>
</table>

**Notes:**
- Each cell of the table has the following structure:
  - **Percentage of recognized patterns**
  - **Mean score of recognized patterns**
  - **Standard Deviation**
- The last column represents the number of patterns to be classified in each Probability Distribution Function.

**Table 1**
5. REFERENCES


