A Novel Approach to Disqualify Datasets Using Accumulative Statistical Spread Map with Neural Networks (ASSM-NN)

Mahmoud Zaki Iskandarani
Faculty of Science and Information Technology, Al-Zaytoonah University of Jordan, Amman, Jordan
Email: m.iskandarani@hotmail.com

Received 30 April 2015; accepted 23 May 2015; published 27 May 2015

Copyright © 2015 by author and Scientific Research Publishing Inc.
This work is licensed under the Creative Commons Attribution International License (CC BY).
http://creativecommons.org/licenses/by/4.0/

Abstract

A novel approach to detect and filter out an unhealthy dataset from a matrix of datasets is developed, tested, and proved. The technique employs a new type of self organizing map called Accumulative Statistical Spread Map (ASSM) to establish the destructive and negative effect a dataset will have on the rest of the matrix if stayed within that matrix. The ASSM is supported by training a neural network engine, which will determine which dataset is responsible for its inability to learn, classify and predict. The carried out experiments proved that a neural system was not able to learn in the presence of such an unhealthy dataset that possessed some deviated characteristics, even though it was produced under the same conditions and through the same process as the rest of the datasets in the matrix, and hence, it should be disqualified, and either removed completely or transferred to another matrix. Such novel approach is very useful in pattern recognition of datasets and features that do not belong to their source and could be used as an effective tool to detect suspicious activities in many areas of secure filing, communication and data storage.

Keywords
Pattern Recognition, Informatics, Neural Networks, Data Mining, Classification, Prediction, Statistics

1. Introduction

In general, many neural networks applications are concerned with analyzing issues related to pattern recognition by using a supervised training method with training datasets. This will achieve and inference relationship between input patterns and output results [1]-[3].

How to cite this paper: Iskandarani, M.Z. (2015) A Novel Approach to Disqualify Datasets Using Accumulative Statistical Spread Map with Neural Networks (ASSM-NN). Intelligent Information Management, 7, 138-151.
http://dx.doi.org/10.4236/iim.2015.73013
Neural Networks offers several algorithms and establishes models using data. Before applying any of the built-in functions for training, it is important to check that the data is reasonable, as good models can be obtained from poor or insufficient data. There is no specific procedure yet that can be used to test the quality or homogeneity and coherence of the data [4]-[6].

One way to check for quality is to view graphical representations of the data in question in the hope of selecting a reasonable subset while eliminating problematic parts. In examining the data for a classification problem, some reasonable points should be looked at, such as:

a) Equal representation of classes by datasets;

b) The presence of dissimilar datasets from the rest or neighboring values.

In supervised training, parameters such as weights and bias matrices for the neural network are used in order to classify all patterns in the training datasets. Larger training datasets are expected to reduce the overall error and error rate. However, it is a challenging task to produce a neural network that will be able to accommodate all patterns in a large training dataset, due to some patterns are difficult to classify. Even if network layers and neurons are modified, there are still some problems in pattern classification despite the lengthy training process [7]-[10].

The probability of occurrence of these patterns is expected to increase as a function of the size of the training dataset. Hence, the neural network will fail to recognize a pattern that approximates to one of the misclassified patterns. Also, if a new pattern is employed, which approximates to one of the misclassified patterns in the old training dataset, the neural network will not be able to classify it, and it will become a new misclassified; thus, the error rate will increase [11]-[14].

In this paper, a new approach catching and isolating such patterns is presented. The approach uses a new Accumulative Statistical Spread Map (ASSM) to initially establish the coherence of the patterns under consideration, and will not cause a misclassification in the neural network, and then when the status is established, the neural structure is used to determine which of the datasets and patterns is causing such misclassification and raising the error rate. Thus, such a neural network structure can be used as a filter and isolator with the Accumulative Statistical Spread Map used to determine which part or parts of the datasets is causing the problem. All the matrices of weights and biases are kept in the order that originally set from the training process throughout the testing process. Moreover, analysis results are used to control the updating process for new patterns.

2. Methodology

Two datasets that belong to the same general parent category are used in the experimental process to prove the concept. Same rules applied to produce the sets; hence, many general common features exist between them. The post processed datasets obtained using the ASSM approach with groups and subgroups produced to show two aspects:

a) If the datasets that the patterns represent belong to the same main group;

b) If there are signs that there will be a conflict in using them together, as one of them has an undesirable effect.

If the result in b is a confirmation, then a neural network algorithm is used to determine which one of the datasets and their patterns is the unhealthy one by carrying out the following steps:

1) Training and testing the considered datasets according to:
   a) Dataset 1;
   b) Dataset 2;
   c) Combined Datasets 1 & 2;

2) Reporting margins of errors covering all three combinations of training;

3) Noticing the pattern that is least affected and classifying it as the unhealthy pattern.

The sorting algorithm is based on the expressions in Equations (1) and (2):

\[
Category_i = f\left(\frac{1}{n} Token_{j}\right)
\]

\[
Input_{ASSM} = \sum_{i=1}^{n} Token_{j}
\]

where
A correlation function between the Tokens;

\( n \): Range of classification;

\( j \): Range of Tokens (in this work \( j = 4 \));

The ASSM carries out initial re-organization and sorting by correlation according to Equation (3), before it produces the final output:

\[
ASSM_{Organization} = Corr\left(\text{Category}_{i}, Input_{ASSM}\right)
\]

(3)

3. Results

Table 1 and Table 2 show the used and categorized Datasets 1 and 2. These Datasets are used in the experimental process.

| Category | Token\(_1\) | Token\(_2\) | Token\(_3\) | Token\(_4\) |
|----------|------------|------------|------------|------------|
| 1348     | 2          | 3          | 1          | 1          |
| 1935     | 4          | 5          | 3          | 4          |
| 2346     | 6          | 4          | 7          | 2          |
| 2470     | 9          | 6          | 2          | 3          |
| 2769     | 1          | 1          | 5          | 18         |
| 3698     | 13         | 8          | 10         | 5          |
| 3815     | 5          | 2          | 9          | 23         |
| 5071     | 11         | 12         | 8          | 8          |
| 5167     | 16         | 9          | 14         | 6          |
| 6099     | 8          | 11         | 12         | 14         |
| 6247     | 19         | 7          | 13         | 13         |
| 6851     | 7          | 10         | 11         | 34         |
| 7367     | 15         | 13         | 15         | 7          |
| 7418     | 14         | 22         | 4          | 10         |
| 8932     | 18         | 19         | 6          | 17         |
| 9190     | 10         | 16         | 20         | 11         |
| 10,405   | 24         | 17         | 21         | 9          |
| 10,934   | 22         | 14         | 17         | 19         |
| 11,527   | 3          | 32         | 35         | 16         |
| 12,024   | 17         | 30         | 16         | 12         |
| 12,854   | 23         | 15         | 23         | 29         |
| 13,000   | 12         | 25         | 36         | 21         |
| 13,381   | 20         | 18         | 22         | 28         |
| 14,056   | 30         | 24         | 28         | 15         |
| 14,242   | 25         | 21         | 18         | 27         |
| 14,336   | 33         | 20         | 32         | 20         |
| 14,636   | 21         | 26         | 19         | 26         |
| 15,747   | 29         | 23         | 30         | 24         |
| 16,215   | 26         | 28         | 27         | 33         |
| 16,280   | 32         | 27         | 24         | 30         |
| 17,255   | 34         | 29         | 25         | 31         |
| 17,817   | 28         | 33         | 26         | 32         |
| 18,087   | 27         | 35         | 31         | 25         |
| 19,032   | 35         | 34         | 29         | 22         |
| 19,180   | 36         | 31         | 33         | 35         |
| 20,362   | 31         | 36         | 34         | 36         |
Table 2. Initial categorization results obtained for dataset 2.

| Category | Token1 | Token2 | Token3 | Token4 |
|----------|--------|--------|--------|--------|
| 915      | 1      | 1      | 1      | 1      |
| 2258     | 3      | 3      | 2      | 3      |
| 2505     | 2      | 2      | 3      | 4      |
| 3754     | 7      | 7      | 5      | 2      |
| 5358     | 4      | 8      | 7      | 5      |
| 6869     | 11     | 6      | 8      | 6      |
| 7131     | 10     | 9      | 4      | 17     |
| 7591     | 6      | 5      | 14     | 38     |
| 8972     | 26     | 4      | 32     | 37     |
| 9519     | 8      | 10     | 10     | 10     |
| 11,305   | 5      | 11     | 26     | 25     |
| 11,775   | 13     | 12     | 11     | 28     |
| 12,222   | 19     | 13     | 17     | 8      |
| 13,227   | 14     | 16     | 21     | 29     |
| 13,240   | 25     | 15     | 12     | 36     |
| 13,245   | 17     | 20     | 9      | 27     |
| 13,313   | 18     | 22     | 22     | 7      |
| 13,786   | 15     | 21     | 13     | 24     |
| 13,933   | 22     | 18     | 20     | 20     |
| 14,233   | 20     | 19     | 38     | 15     |
| 14,356   | 31     | 14     | 27     | 9      |
| 14,379   | 12     | 23     | 37     | 14     |
| 14,879   | 16     | 25     | 16     | 31     |
| 14,993   | 35     | 27     | 6      | 33     |
| 15,250   | 9      | 28     | 31     | 35     |
| 15,718   | 21     | 26     | 39     | 16     |
| 16,511   | 39     | 24     | 24     | 22     |
| 16,700   | 40     | 17     | 15     | 26     |
| 17,639   | 38     | 29     | 18     | 18     |
| 17,700   | 27     | 31     | 30     | 34     |
| 17,769   | 34     | 30     | 35     | 12     |
| 17,829   | 28     | 32     | 36     | 13     |
| 18,029   | 23     | 36     | 33     | 39     |
| 18,499   | 32     | 33     | 25     | 23     |
| 18,735   | 36     | 35     | 23     | 11     |
| 18,742   | 37     | 34     | 34     | 40     |
| 19,016   | 24     | 36     | 40     | 21     |
| 19,243   | 29     | 37     | 29     | 32     |
| 19,942   | 30     | 39     | 28     | 30     |
| 21,144   | 33     | 40     | 19     | 19     |
4. Discussion and Conclusions

Figure 1 and Figure 2 show Accumulative Statistical Spread Map (ASSM) for Datasets 1 and 2. The maps represent the correlated statistical features in both Dataset 1 and Dataset 2 in relation to the sequence numbers assigned to them due to sorting.
Figure 2. ASSM-dataset 2.
From the maps, the following is realized:
1) The concentration of feature spread into the second and fourth quarters of the ASSM (counterclockwise);
2) The similarity in feature spread confirms that the two patterns representing Datasets 1 and 2 derive from similar source with common features.

Each ASSM is processed to produce clustered representation, as shown in Figure 3 and Figure 4, in order to obtain a similar representation to genetic code. This is shown in groups 1 and 2.

Dataset 1 → \{3, 5, 0, 0, 0\}, \{5, 45, 4, 6, 0\}, \{0, 6, 3, 48, 3\}, \{0, 0, 4, 4\}  
Dataset 2 → \{5, 3, 0, 0, 0\}, \{3, 45, 5, 15, 0\}, \{0, 4, 1, 3, 0\}, \{0, 14, 2, 46, 6\}, \{0, 2, 0, 4, 2\}

From groups 1 and 2, the following is deduced:
1) The existence of inverted digits (features position swapping) between dataset 1 and dataset 2; Inverted digit values are expected to have a destructive effect on the system learning and classification process;
2) The presence of common features (features of similar value and position);
3) The presence of different features (features of different values in similar positions);

Placing the code groups into matrices 3 and 4 and carrying out row and column summation, shows the following:
1) Each dataset follows an overall mathematical code that is specific to its representational pattern;
2) The symmetrical relationship between each Row and Column of each matrix. This is the result of using ASSM and indicates that both patterns belong to the same process;

![Figure 3. Clustered representation of dataset 1.](image)

![Figure 4. Clustered representation of dataset 2.](image)
3) The difference in the Column values with higher values appearing in Dataset 2 supports the evidence of Dataset 2 capability to overshadow Dataset 1 and negatively affect the overall learning and classification process.

$$
\begin{bmatrix}
3 & 5 & 0 & 0 & 0 \\
5 & 45 & 4 & 6 & 0 \\
0 & 4 & 1 & 2 & 1 \\
0 & 6 & 3 & 48 & 3 \\
0 & 0 & 0 & 4 & 4
\end{bmatrix}
\rightarrow \text{Row} = \{8, 60, 8, 60, 8\}, \text{Column} = \{8, 60, 8, 60, 8\}

(3)

$$
\begin{bmatrix}
5 & 3 & 0 & 0 & 0 \\
3 & 45 & 5 & 15 & 0 \\
0 & 4 & 1 & 3 & 0 \\
0 & 14 & 2 & 46 & 6 \\
0 & 2 & 0 & 4 & 2
\end{bmatrix}
\rightarrow \text{Row} = \{8, 68, 8, 68, 8\}, \text{Column} = \{8, 68, 8, 68, 8\}

(4)

The previous results indicate the presence of conflicting patterns, where one of them would cause a problem when used with the rest of similar patterns. To uncover the unhealthy pattern responsible for such condition, a back propagation neural algorithm is employed to train and test both datasets.

Table 3 and Table 4 show the results for sorting, categorization, and neural networks training and testing

| Desired Output | Training Input | Actual Output | Sequential Number |
|----------------|----------------|---------------|-------------------|
| Category       | Token\(_1\)   | Token\(_2\)   | Token\(_3\)   | Token\(_4\)   | Category | Category | |
| 1348           | 2              | 3              | 1              | 1              | 1348     | 1        |
| 1935           | 4              | 5              | 3              | 4              | 1936     | 2        |
| 2346           | 6              | 4              | 7              | 2              | 2346     | 3        |
| 2470           | 9              | 6              | 6              | 3              | 2470     | 4        |
| 2769           | 1              | 1              | 5              | 18             | 2769     | 5        |
| 3698           | 13             | 8              | 10             | 5              | 3698     | 6        |
| 3815           | 5              | 2              | 9              | 23             | 3815     | 7        |
| 5071           | 11             | 12             | 8              | 8              | 5070     | 8        |
| 5167           | 16             | 9              | 14             | 6              | 5167     | 9        |
| 6099           | 8              | 11             | 12             | 14             | 6099     | 10       |
| 6247           | 19             | 7              | 13             | 13             | 6247     | 11       |
| 6851           | 7              | 10             | 11             | 34             | 6851     | 12       |
| 7367           | 15             | 13             | 15             | 7              | 7367     | 13       |
| 7418           | 14             | 22             | 4              | 10             | 7418     | 14       |
| 8932           | 18             | 19             | 6              | 17             | 8932     | 15       |
| 9190           | 10             | 16             | 20             | 11             | 9190     | 16       |
| 10,405         | 24             | 17             | 21             | 9              | 10,405   | 17       |
| 10,934         | 22             | 14             | 17             | 19             | 10,934   | 18       |
| 11,527         | 3              | 32             | 35             | 16             | 11,527   | 19       |
| 12,024         | 17             | 30             | 16             | 12             | 12,024   | 20       |
| 12,854         | 23             | 15             | 23             | 29             | 12,853   | 21       |
| 13,000         | 12             | 25             | 36             | 21             | 13,000   | 22       |
| 13,381         | 20             | 18             | 22             | 28             | 13,382   | 23       |
| 14,056         | 30             | 24             | 28             | 15             | 14,056   | 24       |
| 14,242         | 25             | 21             | 18             | 27             | 14,242   | 25       |
| 14,336         | 33             | 20             | 32             | 20             | 14,336   | 26       |
| 14,636         | 21             | 26             | 19             | 26             | 14,636   | 27       |
| 15,747         | 29             | 23             | 30             | 24             | 15,747   | 28       |
| 16,215         | 26             | 28             | 27             | 33             | 16,215   | 29       |
| 16,280         | 32             | 27             | 24             | 30             | 16,279   | 30       |
| 17,235         | 34             | 29             | 25             | 31             | 17,257   | 31       |
| 17,817         | 28             | 33             | 26             | 32             | 17,816   | 32       |
| 18,087         | 27             | 35             | 31             | 25             | 18,088   | 33       |
| 19,032         | 35             | 34             | 29             | 22             | 19,032   | 34       |
| 19,180         | 36             | 31             | 33             | 35             | 19,180   | 35       |
| 20,362         | 31             | 36             | 34             | 36             | 20,362   | 36       |
Table 4. Sorting, training, and testing of dataset 2.

| Desired Output | Training Input | Actual Output | Sequential Number |
|----------------|----------------|---------------|-------------------|
| Category       | Token1 | Token2 | Token3 | Token4 | Category | Sequential Number |
| 915            | 1      | 1      | 1      | 1      | 915      | 1               |
| 2258           | 3      | 3      | 2      | 3      | 2258     | 2               |
| 2505           | 2      | 2      | 3      | 4      | 2503     | 3               |
| 3754           | 7      | 7      | 5      | 2      | 3753     | 4               |
| 5358           | 4      | 8      | 7      | 5      | 5359     | 5               |
| 6869           | 11     | 6      | 8      | 6      | 6870     | 6               |
| 7131           | 10     | 9      | 4      | 17     | 7131     | 7               |
| 7591           | 6      | 5      | 14     | 38     | 7591     | 8               |
| 8972           | 26     | 4      | 32     | 37     | 8972     | 9               |
| 9519           | 8      | 10     | 10     | 10     | 9518     | 10              |
| 11,305         | 5      | 11     | 26     | 25     | 11,305   | 11              |
| 11,775         | 13     | 12     | 11     | 28     | 11,775   | 12              |
| 12,222         | 19     | 13     | 17     | 8      | 12,222   | 13              |
| 13,227         | 14     | 16     | 21     | 29     | 13,228   | 14              |
| 13,240         | 25     | 15     | 12     | 36     | 13,240   | 15              |
| 13,245         | 17     | 20     | 9      | 27     | 13,245   | 16              |
| 13,313         | 18     | 22     | 22     | 7      | 13,313   | 17              |
| 13,786         | 15     | 21     | 13     | 24     | 13,787   | 18              |
| 13,933         | 22     | 18     | 20     | 20     | 13,932   | 19              |
| 14,233         | 20     | 19     | 38     | 15     | 14,233   | 20              |
| 14,356         | 31     | 14     | 27     | 9      | 14,357   | 21              |
| 14,379         | 12     | 23     | 37     | 14     | 14,378   | 22              |
| 14,879         | 16     | 25     | 16     | 31     | 14,879   | 23              |
| 14,993         | 35     | 27     | 6      | 33     | 14,993   | 24              |
| 15,250         | 9      | 28     | 31     | 35     | 15,249   | 25              |
| 15,718         | 21     | 26     | 39     | 16     | 15,719   | 26              |
| 16,511         | 39     | 24     | 24     | 22     | 16,511   | 27              |
| 16,700         | 40     | 17     | 15     | 26     | 16,700   | 28              |
| 17,639         | 38     | 29     | 18     | 18     | 17,639   | 29              |
| 17,700         | 27     | 31     | 30     | 34     | 17,699   | 30              |
| 17,769         | 34     | 30     | 35     | 12     | 17,769   | 31              |
| 17,829         | 28     | 32     | 36     | 13     | 17,828   | 32              |
| 18,029         | 23     | 36     | 33     | 39     | 18,030   | 33              |
| 18,499         | 32     | 33     | 25     | 23     | 18,500   | 34              |
| 18,735         | 36     | 35     | 23     | 11     | 18,735   | 35              |
| 18,742         | 37     | 34     | 34     | 40     | 18,741   | 36              |
| 19,016         | 24     | 36     | 40     | 21     | 19,017   | 37              |
| 19,243         | 29     | 37     | 29     | 32     | 19,244   | 38              |
| 19,942         | 30     | 39     | 28     | 30     | 19,941   | 39              |
| 21,144         | 33     | 40     | 19     | 19     | 21,144   | 40              |
results for similarly processed datasets with the neural engine trained using dataset 1 in isolation from dataset 2, and dataset 2 in isolation from dataset 1 with RED Sequential Numbers pointing towards the records with difference between Actual and Desired data, while Table 5 and Table 6 show the results for the same datasets merged and sorted in relation to each other, and presented to the neural engine for training and testing.

From Table 7 and Table 8, it is deduced that Dataset 2 prediction starts with larger initial error, and stays unaffected after merging with Dataset 2 as a training set. For Dataset 1, the result is almost opposite, as its prediction starts with much smaller error and suffers large error increase after merging with Dataset 2 as a training set. This indicates a marked increase in the level of total pattern destruction and mutation due to effect of Dataset 2. So, Dataset 2 disabled the proper functionality of the Neural Structure and inhibited its performance.

The previous is supported by the following percentage errors appearing in Table 9, where it is clear that Dataset 2 is not affected by Dataset 1, while Dataset 1 is greatly and adversely affected by the presence of Dataset 2.

| Desired Output | Training Input | Actual Output | Sequential Number |
|----------------|----------------|---------------|-------------------|
| Category       | Token₁         | Token₂        | Token₃           | Token₄ | Category       |
| 1348           | 2              | 3             | 1                | 1      | 1296           | 1               |
| 1935           | 4              | 5             | 3                | 4      | 3429           | 2               |
| 2346           | 6              | 4             | 7                | 2      | 4491           | 3               |
| 2470           | 9              | 6             | 2                | 3      | 3457           | 4               |
| 2769           | 1              | 1             | 5                | 18     | 7253           | 5               |
| 3698           | 13             | 8             | 10               | 5      | 8636           | 6               |
| 3815           | 5              | 2             | 9                | 23     | 9810           | 7               |
| 5071           | 11             | 12            | 8                | 8      | 8437           | 8               |
| 5167           | 16             | 9             | 14               | 6      | 11,360         | 9               |
| 6099           | 8              | 11            | 12               | 14     | 12,265         | 10              |
| 6247           | 19             | 7             | 13               | 13     | 12,374         | 11              |
| 6851           | 7              | 10            | 11               | 34     | 8555           | 12              |
| 7367           | 15             | 13            | 15               | 7      | 12,036         | 13              |
| 7418           | 14             | 22            | 4                | 10     | 6917           | 14              |
| 8932           | 18             | 19            | 6                | 17     | 10,442         | 15              |
| 9190           | 10             | 16            | 20               | 11     | 13,997         | 16              |
| 10,405         | 24             | 17            | 21               | 9      | 13,697         | 17              |
| 11,527         | 3              | 32            | 35               | 16     | 16,206         | 19              |
| 12,024         | 17             | 30            | 16               | 12     | 14,843         | 20              |
| 12,854         | 23             | 15            | 23               | 29     | 14,189         | 21              |
| 13,000         | 12             | 25            | 36               | 21     | 15,584         | 22              |
| 13,381         | 20             | 18            | 22               | 28     | 14,495         | 23              |
| 14,056         | 30             | 24            | 28               | 15     | 16,788         | 24              |
| 14,242         | 25             | 21            | 18               | 27     | 15,914         | 25              |
| 14,336         | 33             | 20            | 32               | 20     | 16,770         | 26              |
| 14,636         | 21             | 26            | 19               | 26     | 16,520         | 27              |
| 15,747         | 29             | 23            | 30               | 24     | 17,338         | 28              |
| 16,215         | 26             | 28            | 27               | 33     | 17,574         | 29              |
| 16,280         | 32             | 27            | 24               | 30     | 18,014         | 30              |
| 17,255         | 34             | 29            | 25               | 31     | 18,664         | 31              |
| 17,817         | 28             | 33            | 26               | 32     | 18,961         | 32              |
| 18,087         | 27             | 35            | 31               | 25     | 20,231         | 33              |
| 19,032         | 35             | 34            | 29               | 22     | 20,904         | 34              |
| 19,180         | 36             | 31            | 33               | 35     | 19,395         | 35              |
| 20,362         | 31             | 36            | 34               | 36     | 19,886         | 36              |
Table 6. Sorting, training, and testing of dataset 2 merged with dataset 1.

| Desired Output | Training Input | Actual Output | Sequential Number |
|----------------|----------------|---------------|--------------------|
| Category       | Token₁  | Token₂  | Token₃  | Token₄  | Category  |                |
| 915            | 1       | 1       | 1       | 1       | 915       | 1               |
| 2258           | 3       | 3       | 2       | 3       | 2258      | 2               |
| 2505           | 2       | 2       | 3       | 4       | 2503      | 3               |
| 3754           | 7       | 7       | 5       | 2       | 3753      | 4               |
| 5358           | 4       | 8       | 7       | 5       | 5359      | 5               |
| 6869           | 11      | 6       | 8       | 6       | 6870      | 6               |
| 7131           | 10      | 9       | 4       | 17      | 7131      | 7               |
| 7591           | 6       | 5       | 14      | 38      | 7591      | 8               |
| 8972           | 26      | 4       | 32      | 37      | 8972      | 9               |
| 9519           | 8       | 10      | 10      | 10      | 9518      | 10              |
| 11,305         | 5       | 11      | 26      | 25      | 11,305    | 11              |
| 11,775         | 13      | 12      | 11      | 28      | 11,775    | 12              |
| 12,222         | 19      | 13      | 17      | 8       | 12,222    | 13              |
| 13,227         | 14      | 16      | 21      | 29      | 13,228    | 14              |
| 13,240         | 25      | 15      | 12      | 36      | 13,240    | 15              |
| 13,245         | 17      | 20      | 9       | 27      | 13,245    | 16              |
| 13,313         | 18      | 22      | 22      | 7       | 13,313    | 17              |
| 13,786         | 15      | 21      | 13      | 24      | 13,787    | 18              |
| 13,933         | 22      | 18      | 20      | 20      | 13,932    | 19              |
| 14,233         | 20      | 19      | 38      | 15      | 14,233    | 20              |
| 14,356         | 31      | 14      | 27      | 9       | 14,357    | 21              |
| 14,379         | 12      | 23      | 37      | 14      | 14,378    | 22              |
| 14,879         | 16      | 25      | 16      | 31      | 14,879    | 23              |
| 14,993         | 35      | 27      | 6       | 33      | 14,993    | 24              |
| 15,250         | 9       | 28      | 31      | 35      | 15,249    | 25              |
| 15,718         | 21      | 26      | 39      | 16      | 15,719    | 26              |
| 16,511         | 39      | 24      | 24      | 22      | 16,511    | 27              |
| 16,700         | 40      | 17      | 15      | 26      | 16,700    | 28              |
| 17,639         | 38      | 29      | 18      | 18      | 17,639    | 29              |
| 17,700         | 27      | 31      | 30      | 34      | 17,699    | 30              |
| 17,769         | 34      | 30      | 35      | 12      | 17,769    | 31              |
| 17,829         | 28      | 32      | 36      | 13      | 17,828    | 32              |
| 18,029         | 23      | 36      | 33      | 39      | 18,030    | 33              |
| 18,499         | 32      | 33      | 25      | 23      | 18,500    | 34              |
| 18,735         | 36      | 35      | 23      | 11      | 18,735    | 35              |
| 18,742         | 37      | 34      | 34      | 40      | 18,741    | 36              |
| 19,016         | 24      | 36      | 40      | 21      | 19,017    | 37              |
| 19,243         | 29      | 37      | 29      | 32      | 19,244    | 38              |
| 19,942         | 30      | 39      | 28      | 30      | 19,941    | 39              |
| 21,144         | 33      | 40      | 19      | 19      | 21,144    | 40              |
| Margin (Dataset 1) | Before Combining | After Combining |
|-------------------|------------------|-----------------|
| 0                 | 52               |                 |
| 1                 | 1494             |                 |
| 0                 | 2145             |                 |
| 0                 | 987              |                 |
| 0                 | 4484             |                 |
| 0                 | 4938             |                 |
| 0                 | 5995             |                 |
| 1                 | 3366             |                 |
| 0                 | 6193             |                 |
| 0                 | 6166             |                 |
| 0                 | 6127             |                 |
| 0                 | 1704             |                 |
| 0                 | 4669             |                 |
| 0                 | 501              |                 |
| 0                 | 1510             |                 |
| 0                 | 4807             |                 |
| 0                 | 3292             |                 |
| 0                 | 2883             |                 |
| 0                 | 4679             |                 |
| 0                 | 2819             |                 |
| 1                 | 1335             |                 |
| 0                 | 2584             |                 |
| 1                 | 1114             |                 |
| 0                 | 2732             |                 |
| 0                 | 1672             |                 |
| 0                 | 2434             |                 |
| 0                 | 1884             |                 |
| 0                 | 1591             |                 |
| 0                 | 1359             |                 |
| 1                 | 1734             |                 |
| 2                 | 1409             |                 |
| 1                 | 1144             |                 |
| 1                 | 2144             |                 |
| 0                 | 1872             |                 |
| 0                 | 215              |                 |
| 0                 | 476              |                 |
Table 8. Margin comparison dataset 2.

| Margin (Dataset 2) | Before Combining | After Combining |
|--------------------|------------------|-----------------|
| 0                  | 0                | 0               |
| 0                  | 0                | 0               |
| 2                  | 2                | 2               |
| 1                  | 1                | 1               |
| 1                  | 1                | 1               |
| 0                  | 0                | 0               |
| 0                  | 0                | 0               |
| 0                  | 0                | 0               |
| 1                  | 1                | 1               |
| 0                  | 0                | 0               |
| 0                  | 0                | 0               |
| 0                  | 0                | 0               |
| 1                  | 1                | 1               |
| 0                  | 0                | 0               |
| 0                  | 0                | 0               |
| 0                  | 0                | 0               |
| 1                  | 1                | 1               |
| 1                  | 1                | 1               |
| 0                  | 0                | 0               |
| 1                  | 1                | 1               |
| 1                  | 1                | 1               |
| 0                  | 0                | 0               |
| 0                  | 0                | 0               |
| 1                  | 1                | 1               |
| 1                  | 1                | 1               |
| 0                  | 0                | 0               |
| 0                  | 0                | 0               |
| 1                  | 1                | 1               |
| 1                  | 1                | 1               |
| 0                  | 0                | 0               |
| 1                  | 1                | 1               |
| 0                  | 0                | 0               |
| 1                  | 1                | 1               |
| 1                  | 1                | 1               |
| 1                  | 1                | 1               |
| 0                  | 0                | 0               |
| 1                  | 1                | 1               |
| 1                  | 1                | 1               |
| 1                  | 1                | 1               |
| 1                  | 1                | 1               |
| 0                  | 0                | 0               |
| 1                  | 1                | 1               |
| 1                  | 1                | 1               |
| 1                  | 1                | 1               |
| 1                  | 1                | 1               |
| 0                  | 0                | 0               |
| 1                  | 1                | 1               |
| 1                  | 1                | 1               |
| 1                  | 1                | 1               |
| 1                  | 1                | 1               |
| 0                  | 0                | 0               |
| 1                  | 1                | 1               |
| 1                  | 1                | 1               |
| 1                  | 1                | 1               |
| 1                  | 1                | 1               |
| 0                  | 0                | 0               |
| 1                  | 1                | 1               |
| 1                  | 1                | 1               |
| 1                  | 1                | 1               |
| 1                  | 1                | 1               |
Table 9. Margin change for datasets 1 and 2.

| Percentage Error (Dataset 1) | Percentage Error (Dataset 2) |
|-----------------------------|-----------------------------|
| Before Combining            | After Combining              |
| 19.4                        | 100                         |
| 50                          | 50                          |

| Percentage Increase in Error |
|-------------------------------|
| 79.6                          |

In conclusion, the ASSM proved its capability to detect and filter out undesirable datasets, which would greatly assist optimizing the neural network structure. Such functionality is critical in facilitating both good neural network designs and isolating certain datasets to study their behavior and reach a conclusion regarding the causes behind such abnormalities. The ASSM can be very useful in Health and Medical applications, such as cases where tumors are involved.

References

[1] Albarado, K., Ledlow, T. and Hartfield, R. (2015) Alternative Analysis Networking: A Multicharacterization Algorithm. *Computing in Science & Engineering*, 17, 54-63. http://dx.doi.org/10.1109/MCSE.2015.10

[2] Li, W., Amsaleg, L., Morton, A. and Marchand-Maillet, S. (2015) A Privacy-Preserving Framework for Large-Scale Content-Based Information Retrieval. *IEEE Transactions on Information Forensics and Security*, 10, 152-167. http://dx.doi.org/10.1109/TIFS.2014.2365998

[3] Cheng, Q., Zhou, H.B., Cheng, J. and Li, H.Q. (2014) A Minimax Framework for Classification with Applications to Images and High Dimensional Data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36, 2117-2130. http://dx.doi.org/10.1109/TPAMI.2014.2327973

[4] Xu, J.L., Ramos, S., Vazquez, D. and Lopez, A.M. (2014) Domain Adaptation of Deformable Part-Based Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36, 2367-2380. http://dx.doi.org/10.1109/TPAMI.2014.2327973

[5] Thanh, M.N., Wu, Q.M.J. and Zhang, H. (2015) Asymmetric Mixture Model with Simultaneous Feature Selection and Model Detection. *IEEE Transactions on Neural Networks and Learning Systems*, 26, 400-4008. http://dx.doi.org/10.1109/TNNLS.2014.2314239

[6] Araujo, A. and Soares, M. (2015) Weights Based Clustering in Data Envelopment Analysis Using Kohonen Neural Network: An Application in Brazilian Electrical Sector. *IEEE—Latin American Transactions*, 13, 188-194. http://dx.doi.org/10.1109/TLA.2015.7040647

[7] Sabri, A. (2014) Further Analysis of Stability of Uncertain Neural Networks with Multiple Time Delays. *Advances in Difference Equations*, 41, 1-16.

[8] Shabtai, A., Moskovitch, R., Feher, C., Dolev, S. and Elovici, Y. (2012) Detecting Unknown Malicious Code by Applying Classification Techniques on OpCode Patterns. *Security Informatics*, 1, 1-22. http://dx.doi.org/10.1186/2193-8532-1-1

[9] Zuech, R., Taghi, T.M. and Wald, R. (2015) Intrusion Detection and Big Heterogeneous Data: A Survey. *Journal of Big Data*, 2, 1-41. http://dx.doi.org/10.1186/s40537-015-0013-4

[10] Goyal, R., Chandra, P. and Singh, Y. (2013) Identifying Influential Metrics in the Combined Metrics Approach of Fault Prediction. *SpringerPlus*, 2, 1-8. http://dx.doi.org/10.1186/2193-1801-2-627

[11] Bashiri, M., Farshbaf-Geranmayeh, A. and Mogouie, H. (2013) A Neuro-Data Envelopment Analysis Approach for Optimization of Uncorrelated Multiple Response Problems with Smaller the Better Type Controllable Factors. *Journal of Industrial Engineering International*, 9, 1-10.

[12] Schiezaro, M. and Pedrini, H. (2013) Data Feature Selection Based on Artificial Bee Colony Algorithm. *EURASIP Journal on Image and Video Processing*, 2013, 1-8. http://dx.doi.org/10.1186/1687-5281-2013-47

[13] Chen, J., Takiguchi, T. and Ariki, Y. (2015) A Robust SVM Classification Framework Using PSM for Multi-Class Recognition. *EURASIP Journal on Image and Video Processing*, 2015, 1-12.

[14] Dimou, I. and Zervakis, M. (2013) On the Analogy of Classifier Ensembles with Primary Classifiers: Statistical Performance and Optimality. *Journal of Pattern Recognition Research*, 8, 98-122. http://dx.doi.org/10.13176/11.497