Assessment Methods of Neural Network Classification Applied to Otoneurological Data

Martti Juhola, Markku Siermala

Department of Computer Sciences, University of Tampere, Finland
Institute of Medical Technology, University of Tampere, Finland

Abstract

Neural networks are applied, among other things, to classify medical data to various disease classes. It is a great advance if data relevance and complicated relations between input data and output values (classifications) can thoroughly be analyzed. Nevertheless, basic techniques for neural network algorithms commonly disallow such chances. We developed methods to analyze such relevance and relations in detail and to outline the entire variable space formed by variables of a dataset and tested these methods with otoneurological data of vertiginous patients. We also treat the problem of using neural network classification in the case of a relatively small dataset that has a biased class distribution, as is usual for medical datasets. To overcome the problem we designed a set of multilayer perceptron neural networks instead of employing only one. This approach improved our classification results at least 10% compared to our previous results with a single network. Our methods are of general type.

Keywords:
Neural networks, Multilayer perceptrons, Variable evaluation, Otoneurological data

1. Introduction

Neural networks are often used as a black-box algorithm to classify medical data. In numerous studies, e.g. [1-3], they were successfully applied to medical problems usually without any precise analysis between input data and output classes. It would be useful to express and to understand variable relevance and relations between input data and classification decisions generated in conjunction with multilayer perceptron networks. We developed novel assessment methods for this purpose and tested them with otoneurological data concerning vertiginous diseases. Previously, we found that it was difficult to apply neural networks to our otoneurological dataset that suffered from a relatively small number of cases and a biased distribution between the disease classes [4,5], which is common for medical datasets. A network tended to find only the majority class. Using the principal component analysis [5] we improved recognition and prediction accuracies, but after this transform it was not possible to explore the afore-mentioned relevance and relations.

We designed methods called scattering, spectrum and response analysis for neural networks
to evaluate variable relevance and relations between the input variables of a multilayer perceptron network and output classes [6]. The approach in [6] was meant for binary coded data of amino acid sequences. Therefore, we now extend it also to other variable types.

2. Materials

The dataset of six disease classes consisted of 815 patient cases, whose diagnoses were made by otoneurological specialists at the Department of Otorhinolaryngology, Helsinki University Central Hospital, Finland. There were patients: 130 (16 % of all) vestibular schwannoma, 146 (18 %) benign positional vertigo, 313 (38 %) Menière’s disease, 41 (5 %) sudden deafness, 65 (8 %) traumatic vertigo, and 120 (15 %) vestibular neuritis cases. When the Menière’s disease subset was far larger than the others and the sudden deafness subset was the least group, the dataset was strongly biased and a single multilayer perceptron tended to detect the former, but to neglect the latter.

The dataset included about 150 variables, but a minority was known to most patients. In our previous work [4,5] we found a subset of important 38 variables and confirmed five of them as the most crucial variables. These subsets were assessed as important also by the physicians, who collected the data. There were 16 quantitative, 10 ordinal, 11 binary variables, and one four-valued nominal variable. The dataset consisted of 11 % missing values imputed with class modes for the nominal (binary) variables and class medians for others, since networks require complete input data.

3. Use of a single neural network

When earlier on [4] all the 38 input variables were utilized, the multilayer perception learned just the majority class. Using only the five most crucial variables it was possible to detect about 75 % of cases on an average. We programmed the networks with Matlab software and found a network of six hidden nodes as best when there were 5 (or 38) input nodes and six output nodes (disease classes). According to crossvalidation, 10 % of the cases were randomly selected to a test set and the rest (90 %) to the corresponding learning set. The selection followed the class distribution of the whole dataset. Ten disjoint test sets (each 10 % of all) were run after training. Ten different runs were accomplished for every test battery. Thus, we ran 100 tests all in all. Means were calculated for accuracies constructed with the subsequent formulas. The recognition accuracy or sensitivity is

\[ r = \frac{tp}{tp + fn} \times 100 \% \]  \hspace{1cm} (1)

and the prediction accuracy or positive predictive value is equal to

\[ p = \frac{tp}{tp + fp} \times 100 \% \]  \hspace{1cm} (2)

where \( tp, fn \) and \( fp \) equal numbers of true positive, false negative and false positive classifications.

4. Use of a set of neural networks

In order to use all the 38 variables and to overcome obstacles borne by the biased disease class distribution we devised the following approach to use a set of six neural networks, one for each disease class. We selected the cases of the corresponding disease for each network. As many as them, we randomly generated artificial cases uniformly distributed over the 38-dimensional variable space outside the area of the cases of the current disease. While
partitioning the disease cases and their “complement” artificial cases into two separate areas, the new distribution was advantageous to the neural network classification. The artificial cases were merely created to map the “complement area” of the current disease in the variable space. The neural network learns such a mapping of the space into two separate areas that entirely cover the space. Every test case was run for all the six networks: the one with the highest output determined a classification. A network with two classes (a disease and its complement area) compensated the original imbalanced distribution between the six disease classes of considerably different sizes, i.e. numbers of cases.

Before any test run, the team of the otoneurological specialists perused all the 815 cases and determined their diagnoses (about 300 cases, in addition to 815, were discarded as too indefinite diagnoses). Average results of each class are given in Table 1, when each of the six networks consisted of only one hidden node. One hidden node was sufficient, when each network included two classes: the disease class and its complement area. The average accuracies were circa 85 %, which is 10 % higher than our earlier results of the basic neural network structures.

<table>
<thead>
<tr>
<th>Disease class</th>
<th>Recognition accuracy [%]</th>
<th>Prediction accuracy [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>vestibular schwannoma</td>
<td>76</td>
<td>85</td>
</tr>
<tr>
<td>benign positional vertigo</td>
<td>81</td>
<td>60</td>
</tr>
<tr>
<td>Menière’s disease</td>
<td>73</td>
<td>82</td>
</tr>
<tr>
<td>sudden deafness</td>
<td>100</td>
<td>89</td>
</tr>
<tr>
<td>traumatic vertigo</td>
<td>92</td>
<td>87</td>
</tr>
<tr>
<td>vestibular neuritis</td>
<td>86</td>
<td>93</td>
</tr>
</tbody>
</table>

5. Scattering of data in the variable space

Scattering was used to map the overlapping of the classes in the variable space. The variable values were scaled into the interval [0,1]. An initial case was randomly chosen from the dataset and its nearest case was determined by the Euclidean distance. The initial case was then discarded. If there were more than one nearest case at the same distance, a move among them was randomly chosen. Thus, a new nearest case was taken and the preceding one dropped out. A counter $a$ was incremented only if the new case was from a different class. The whole dataset was handled in this way, and the ultimate value of $a$ was equal to a total number of changes between the classes.

The scattering ratio $s$ describes positioning of cases and is applied to k classes with the equation

$$s = \frac{a - (k - 1)}{A},$$

where $A$ is the maximum number of changes as follows. Let $m_G$ be the number of the cases in the greatest class. If it is greater than the sum $M_O$ of the numbers of the cases in all other classes, then $A$ is equal to $2M_O$. Otherwise $A$ is equal to $M-1$, where $M$ is the number of the cases in all the classes.

The less the scattering ratio is, the more appropriately the cases are located in the variable space. The ratio would be equal to 1 if the cases were perfectly alternately selected from different classes. If the cases were fully randomly located in the variable space, the ratio would be equal to
where \( p_i \) is the proportional frequency of the class \( i \), \( c_i \) the case number in \( i \), \( C \) the class number, and \( N \) the number of all cases. For the six classes and 815 cases, this bound was 0.77. An average ratio was produced from 20 runs, since they were slightly different due to various initializations. When ratio 0.19 was closer to 0 than 0.77, it referred to the good separation of the classes. Variables (2), (3), (4), and (11) of the five variable subset were very essential in the scattering analysis (Fig. 1).

\[
s_{random} = 1 - \frac{\sum_{i=1}^{C} p_i c_i}{N - 1},
\]

6. Spectrum of neural network outputs

Our first spectrum technique [6] was a multivariate model, but we altered to a univariate one, since a neural network variously behaves after different learning times. A random initialization alters its weights, and thus the network does not recognize exactly the same features from the dataset from one test to another. All variables were scaled to the interval \([0,1]\). A network was trained for one variable at a time. For each class the spectrum was evolved inputting every discretized value of a variable and computing its outcome value. We computed six curves for the classes. Fig. 2 represents variable (1): patient’s age at the first spell. This illustrates that if the first spell appears around the age of ten years, it strongly refers to vestibular neuritis (a large bump in the lowest curve).

7. Response of neural network outputs

We designed the response algorithm to reveal relations between input variables and output classes:

1. Generation: All input combinations are evolved to the neural network and such combinations are chosen, which give high responses for the treated output class.
2. Switching off: Every variable of a combination detected in step 1 is considered. The other variables are temporarily held fixed. The current variable is perused throughout its all possible input values, and on the basis of them the corresponding output values are computed. We check if output values sufficiently vary depending on a threshold value. Otherwise, the variable is unimportant to this combination and, and the variable remains passive. It is switched off.
3. Clustering: Local clusters are formed from the set of the combinations computed. The centre case computed from the cluster is due to represent the cluster.

4. Interpretation: The disease class is interpreted by using the centre cases.

Figure 2 - Spectrum of the variable (1) for the classes (from top): vestibular schwannoma, benign positional vertigo, Menière’s disease, sudden deafness, traumatic vertigo, and vestibular neuritis

The neural network normally detects several inputs typical to the underlying disease class. When there are many local optima, there are several different interesting combinations that ought to be processed. In the response analysis a genetic algorithm was employed and it was allowed to get stuck to local optima to reveal complicated properties of the dataset.

If the disease profile is complicated, there may be several inputs to yield high responses. There may be identical, quite similar, or different combinations. Identical and quite similar combinations are clustered together, and a centre case is evaluated for them. A centre case is the interpretation of the network about input combinations induced by a disease class.

To find local optima, a genetic algorithm was run with 100 generations and a population of 200 elements. The best case was determined as an output node of a disease class, which gave the greatest difference between its own output value and the mean of the other output nodes. Ten variable combinations with strong responses were evolved along with step 1 for every class. Weak variables were switched off from these variable combinations in step 2. If the value of the output node did not exceed a given switching off threshold, the variable was deemed to be unnecessary for the current variable combinations in the disease class. Table 2 presents response analysis results. The variables rated unnecessary for combinations were given as empty elements or omitted lines. The non-empty elements correspond to the original non-scaled variable values, but continuous variables as discrete.

Any fairly similar cases were clustered together by step 3. Two cases were seen similar if the value of one variable differed less than 10 % from that of the other. Their numbers are shown at the lowest row in Table 2. We used a switching off threshold value of 0.7. For example, the variable (23) indicated that the value of 0 is required to be in one cluster, i.e. the value of this binary variable is false or no (spontanic nystagmus) for Menière’s disease.

8. Discussion

Applying a set of six neural networks, one for each disease class, we could improve recognition and prediction accuracies and obtain at least as good results as using, e.g. principal component analysis in our earlier research. In a way the present method is superior to those earlier, since we treated merely the original variables, not using any transforms. By inputting the original variables to the neural networks allows to study relations between input
variables and output classes. The significant hardship incurred by the biased class distribution was discarded with this approach of the set of several networks. Note that this method is independent of data used.

We presented three assessment methods for multilayer perceptron neural networks. The spectrum expresses the influence of individual variables on the disease classes and compels the neural network to compute the strongest suggestion from a suspected disease to every variable value. The response analysis widens the former investigation of variable influence on the disease classes. These results are useful both for machine learning research and for otoneurologists.

### Table 2 – Response analysis results of Menière’s disease

<table>
<thead>
<tr>
<th>Variable</th>
<th>Clusters: variable values</th>
<th>Explanation of value</th>
</tr>
</thead>
<tbody>
<tr>
<td>(14) hearing loss type</td>
<td>1</td>
<td>sudden hearing loss</td>
</tr>
<tr>
<td>(23) spontaneous nystagmus</td>
<td>0</td>
<td>no spontaneous nystagmus</td>
</tr>
<tr>
<td>(24) swaying velocity eyes open</td>
<td>8</td>
<td>&gt;1.5 cm/s; poor posture upkeep</td>
</tr>
<tr>
<td>(25) swaying velocity eyes closed</td>
<td>7</td>
<td>&gt;2.1 cm/s; poor posture upkeep</td>
</tr>
<tr>
<td>(26) spontaneous nystagmus (eye)</td>
<td>7 6</td>
<td>&gt; 0 °/s; abnormal</td>
</tr>
<tr>
<td>(28) nystagmus to right (eye velocity)</td>
<td>23 33</td>
<td>&gt; 20 °/s; strong response</td>
</tr>
<tr>
<td>(29) nystagmus to left (eye velocity)</td>
<td>23 14 40</td>
<td>&gt; 20 °/s; strong response</td>
</tr>
<tr>
<td>(38) sensation of lightheadedness</td>
<td>1 0 0 0 0 0</td>
<td>0 = no, 1 = yes;</td>
</tr>
<tr>
<td>Number of value combinations</td>
<td>1 1 1 2 5</td>
<td></td>
</tr>
</tbody>
</table>

### 9. Acknowledgments

The authors are grateful to E. Kentala, M.D., and prof. I. Pyykkö, M.D., for medical consultation in respect of otoneurological data and its acquisition, and J. Laurikkala, Ph.D., for statistical aid.

### 10. References


**Address for correspondence**

Martti Juhola, Department of Computer Sciences, FL-33014 University of Tampere, Martti.Juhola@cs.uta.fi

---

*Section 3: Decision Support and Clinical Guidelines*