A machine learning approach to differentiating bacterial from viral meningitis.

Kenneth Revett[1]
Florin Gorunescu[2]
Marina Gorunescu[3]
Marius Ene[2]

[1] Harrow School of Computer Science, University of Westminster
[2] Department of Mathematics, Biostatistics and Computer Science, University of Medicine and Pharmacy of Craiova
[3] Department of Computer Science, Faculty of Mathematics and Computer Science, University of Craiova

A Machine Learning Approach to Differentiating Bacterial From Viral Meningitis

Revett, K., Gorunescu F., Gorunescu M., & Ene M.

Abstract -- Clinical reports indicate that differentiating bacterial from viral (aseptic) meningitis is still a difficult issue, compounded by factors such as age and time of presentation. Clinicians routinely rely on the results from blood and cerebrospinal fluid (CSF) to discriminate bacterial from viral meningitis. Tests such as the CSF Gram stain performed prior to broad-spectrum antibiotic treatment yield sensitivities between 60 and 92%. Sensitivity can be increased by performing additional laboratory testing, but the results are never completely accurate and are not cost effective in many cases. In this study, we wished to determine if a machine learning approach, based on rough sets and a probabilistic neural network could be used to differentiate between viral and bacterial meningitis. We analysed a clinical dataset containing records for 581 cases of acute bacterial or viral meningitis. The rough sets approach was used to perform dimensionality reduction in addition to classification. The results were validated using a probabilistic neural network. With an overall accuracy of 98%, these results indicate rough sets is a useful approach to differentiating bacterial from viral meningitis.

Keywords: automated diagnosis, meningitis, probabilistic neural networks, and rough sets

K Revett is with the Harrow School of Computer Science, University of Westminster, London, UK (phone: 44-2079115000; fax: 44-2079115609; e-mail: revettk@westminster.ac.uk).

F. Gorunescu, is with the Department of Mathematics, Biostatistics and Computer Science, University of Medicine and Pharmacy of Craiova, Romania (e-mail: fgorun@rdslink.ro); *IEEE member*

M. Gorunescu is with the Department of Computer Science, Faculty of Mathematics and Computer Science, University of Craiova, Romania (e-mail: mgorun@inf.ucv.ro)

M. Ene is with the Department of Mathematics, Biostatistics and Computer Science, University of Medicine and Pharmacy of Craiova, Romania (e-mail: enem@umfcv.ro)

1. INTRODUCTION

Meningitis is an infection of tissue that forms a protective covering of the surface of the brain (the meninges). The primary cause is due to either bacterial or viral infection. The primary symptoms of meningitis are: high fever, lethargy, vomiting, upper respiratory symptoms, seizure, and petechia [1]. Though not as common as it once was, it is still a disease that can be fatal if left untreated. Meningitis is a disease with a high mortality rate – which ranges from 10-30%, despite advances in treatment estimates indicate that 5-40% of the survivors suffer permanent hearing deficiencies [2]. With the advent of vaccines such as the *Haemophilus influenzae* type b (introduced in 1990), the incidence has declined, with a reduction of up to 94% in the number of new cases of H. *influenzae* meningitis (bacterial). In 2000, the conjugate pneumococcal vacine was introduced, which further reduced the number of new cases reported. Yet despite these medical advances, in the United States alone, there are on average 6,000 new cases of bacterial meningitis diagnosed per annum, half of which affect people under the age of 18 [3]. With regards to viral meningitis (the major form of aseptic meningitis) 36,000 hospitalisations and 175,000 hospital days occur annually in the United States alone, yielding an estimated annual cost of between $234 and $310 million [4].

The clinical outcome for bacterial meningitis is much graver than the aseptic (viral) form of the disease. When a patient is presented to an emergency healthcare facility, the patient receives a broad-spectrum antibiotic when meningitis is suspected. The next stage in the diagnostic procedure entails bacterial cultures, a CT scan and lumbar puncture if the CT scan s negative. Generally, physicians rely on the results of the bacterial culture (specifically the Gram stain) to differentiate between bacterial and viral meningitis. In addition, other parameters such as CSF white blood cell count(WBC), blood and CSF glucose levels,

IEEE John Vincent Atanasoff 2006 International Symposium on Modern Computing (JVA'06)
0-7695-2643-8/06 $20.00 © 2006 IEEE

polymorhonuclear (PMN) cells, patient age, month of the year (i.e. summer months are termed the 'entervirus season'), CSF protein, and CSF leukocyte count among others are measured as clinically relevant to diagnosis. Unfortunately, these measurements are not always performed due to staff shortages, or are performed after a delay after hospitalization which may alter the values of these parameters to such an extent that their values are no longer indicative of the diagnosis – at least without taking the time factor into account.

Several attempts to produce statistical based models have published [2,3,4] using multivariate statistical and recursive partitioning models. These results have proven successful in producing accurate classification results – but are highly dependent on the dataset available and are laborious – in that the variables are analysed in a univariate fashion and then combined into a multi-variate model either directly through logistic analysis or through recursive partitioning. In this paper, we have applied the rough sets paradigm to the study of a rather large but incomplete dataset of patients that were diagnosed with either bacterial or aseptic (viral) meningitis. We propose that rough sets has the capacity to generate multivariate models without the need to analyse each variable in a univariate fashion. In addition, rough sets is able to generate a set of decision rules which have a very readable form that can be directly used by clinicians. Lastly, the classification accuracy of rough sets on small biomedical datasets has been demonstrated to be as accurate if not more so than most 'traditional' methods [5,6]. In order to validate the results we derived from our rough sets analysis, we use a probabilistic neural network (PNN). PNN have the ability to perform very fast classification and has been used successfully on a number of small biomedical datasets. In the next section, we present an overview of the PNN and rough sets methodology, along with a brief description of the dataset. Next we present some of the key results of this work, followed by a brief discussion of the main conclusions that can be dawn from this research.

2. METHODS

2.1 ROUGH SETS

Our hepatic cancer diagnosis pre-processing/classifier is based on the concept of approximate reducts derived from the data-mining paradigm of the theory of Rough Sets [7],[8]. We divide the table into training and test cases, employing N-fold cross validation. The data set is transformed into a decision table (DT) from which rules are generated to provide an automated classification capacity. In generating the decision table, each row consists of an observation (also called an object) and each column is an attribute, with the last one as the decision for this object {d}. Formally, a DT is a pair A = (*U,* A∪{d}) where d □ A is the *decision attribute,* where *U* is a finite non-empty set of objects called the *universe* and A is a finite non-empty set of attributes such that $a:U->V_a$ is called the value set of a. Rough sets seeks data reduction through the concept of equivalence classes (through the indiscernibiliy relation). By generating such classes, one can reduce the number of attributes in the decision table by selecting any member of the equivalence class as a representation of the entire class. This process generates a series of *reducts* – which are subsequently used in the classification process. Finding the reducts is an NP-hard problem, but fortunately there are good heuristics that can compute a sufficient amount of approximate reducts in reasonable time to be usable. In the software system that we employ an order based genetic algorithm (o-GA) ([9]) is used to search through the decision table for approximate reducts which result in a series of 'if..then..' decision rules. We then apply these decision rules to the test data and measure specificity and sensitivity of the resulting classifications. In addition, we examined in a systematic fashion, which attributes were most informative in the decision process – this can be determined by examining the correlation, coverage, and support of the attributes in the final set of decision rules. This provides us with the entry point for using the probabilistic neural network approach – as a corroborative technique in his particular experiment, which we describe in the next section.

2.2 PROBABILISTIC NEURAL NETWORKS

The PNNs are basically classifiers. The general classification problem is to determine the category membership of a multivariate sample data (i.e. a *p*-dimensional random vector **x**) into one of *q* possible groups Ω_i, $i = 1, 2,\ldots, q$,

based on a set of measurements. If we know the probability density functions (p.d.f.) $f_i(\mathbf{x})$, usually the Parzen-Cacoulos or Parzen like p.d.f. classifiers:

$$f_i(x) = \frac{1}{(2\pi)^{p/2}\sigma^p} \cdot \frac{1}{m_i} \cdot \sum_{j=1}^{m_i} \exp\left(-\frac{\left\| x - x_j \right\|^2}{2\sigma^2} \right), \quad (1)$$

the *a priori* probabilities $h_i = P(\Omega_i)$ of occurrence of patterns from categories Ω_i and the *loss* (or *cost*) parameters l_i associated with all incorrect decisions given $\Omega = \Omega_i$, then, according to the Bayesian decision rule, we classify $\mathbf{x}$ into the category Ω_i if the inequality $l_i\ h_i\ f_i(\mathbf{x}) > l_j\ h_j\ f_j(\mathbf{x})$ holds true. The standard training procedure for PNN requires a single pass over all the training patterns, giving them the advantage of being faster than the feed-forward neural networks [10].

Basically, the architecture of PNN is limited to three layers: the *input/pattern layer*, the *summation layer* and the *output layer*. Each input/pattern node forms a product of the input pattern vector $\mathbf{x}$ with a weight vector W_i and then perform a nonlinear operation, that is $\exp[-(W_i - x)^\tau (W_i - x)/(2\sigma^2)]$ (assuming that both $\mathbf{x}$ and W_i are normalized to unit length), before outputting its activation level to the summation node. Each summation node receives the outputs from the input/pattern nodes associated with a given class and simply sums the inputs from the pattern units that correspond to the category from which the training pattern was selected, $\sum_i \exp[-(W_i - x)^\tau (W_i - x)/(2\sigma^2)]$. The output nodes produce binary outputs by using the inequality:

$$\sum_i \exp[-(W_i - x)^\tau (W_i - x)/(2\sigma^2)] > \sum_j \exp[-(W_j - x)^\tau (W_j - x)/(2\sigma^2)], \quad (2)$$

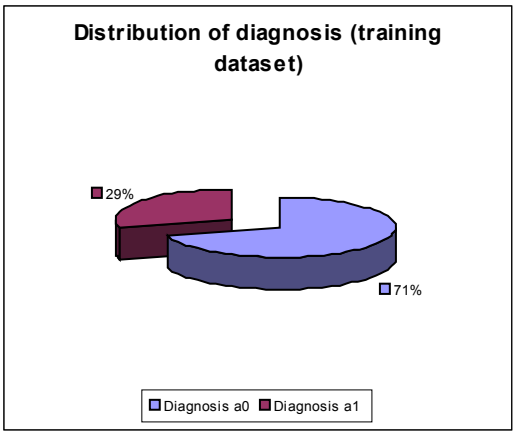
related to two different categories Ω_i and Ω_j. The key to obtain a good classification using PNN is to optimally estimate the two parameters of the Bayes decision rule, the misclassification costs and the prior probabilities. In our practical experiment we have estimate them heuristically. Thus, as concerns the costs parameters, we have considered them depending on the average distances D_i, inversely proportional, that is $l_i = 1/D_i$. As concerns the prior probabilities, they measure the membership probability in each group and, thus, we have considered them equal to each group size, that is $h_i = m_i$. As in our previous work, we employed an evolutionary technique based on the genetic algorithm to find the smoothing parameters (cf [11] for implementation details). To avoid overfitting, the data set was randomly partitioned into two sets: the training set and the validation set. A number of 458 persons (70%) of the initial group were withheld from the initial group for the smoothing factor adjustment (the training process). Once optimal smoothing parameters σ's for each decision category were obtained using the training set, the trained PNN was applied to the validation set (the remaining 241 persons).

2.3 DECISION TABLE DESCRIPTION

The dataset that was used in this study consisted on 581 records with 21 attributes for each. Unfortunately, many of the objects contains missing values – yielding a 'swiss cheese' affect to the dataset. In all, there were a total of 110 objects that were complete. The attributes contained a mixture of continuous and discrete data items and there were 2 decision classes: bacterial or viral meningitis. The data was derived in a retrospective study from patients that were hospitalised between January 1969 and July 1980. For a complete listing of the 21 attributes see reference [2] for details.

3. RESULTS

For the rough sets based approach, several steps are required in order to classify the data. The first stage with this dataset was to use imputation or not. We employ a conditioned mean/mode imputation algorithm when imputation is used throughout this study. The dataset contained a significant number of missing values – that were randomly spread across attributes. When imputation was used, we obviously were able to use the full complement of objects found in the dataset – 581. The next stage was to then discretise those attributes that contained continuous values. This was performed using an entropy/MDL algorithm. Next, a partitioning of the data into a test and training set was performed – we tried various splits: 50/50, 60/40, 75/25 and 80/20 in this work. We performed 5-fold cross validation of the data, in order to enhance the statistical validity of the



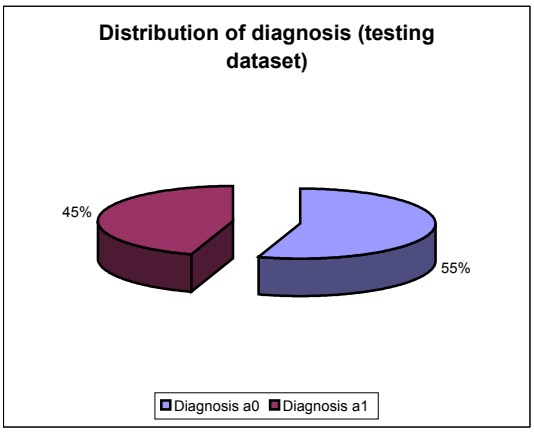


Figure 1. This figure displays the diagnosis distributions for the training and testing set used For the PNN experiments. Please note the diagnosis a0 represents viral and a1 represents bacterial meningitis.

Figure 2a,b) Statistics of the attributes employed in the testing (a) and training (b) data for the reduced rough sets and the PNN algorithms.

TRAINING DATASET	Mean	Confidence -95%	Confidence +95%	Std.Dev.
Age	18.46	13.13	23.79	21.49
Bloodgl	129.07	117.41	140.74	47.07
Gl	59.86	53.03	66.69	27.56
Pr	147.55	80.57	214.52	270.29
Phys	54.40	45.52	63.27	35.82
Lymphs	53.00	35.07	70.92	72.32

TESTING DATASET	Mean	Confidence -95%	Confidence +95%	Std.Dev.
Age	18.46	13.13	23.79	21.49
Bloodgl	129.07	117.41	140.74	47.07
Gl	59.86	53.03	66.69	27.56
Pr	147.55	80.57	214.52	270.29
Phys	54.40	45.52	63.27	35.82
Lymphs	53.00	35.07	70.92	72.32

Figure 3) Statistical analysis of the attributes comparing between groups using the standard student t test.

t-test for independent samples	t-value	df	p	F-ratio	P
Age (training) vs. Age (testing)	-0.59	114	0.55	1.17	0.53
Bloodgl (training) vs. Bloodgl (testing)	-1.78	114	0.07	2.18	0.003
Gl (training) vs. Gl (testing)	-0.10	114	0.91	1.84	0.02
Pr (training) vs. Pr (testing)	0.14	114	0.88	3.18	0.00
Phys (training) vs. Phys (testing)	-0.24	114	0.80	1.14	0.59
Lymphs (training) vs. Lymphs (testing)	0.99	114	0.32	3.74	0.00

results. For the PNN approach, we used only those objects and attributes that formed the largest complete set. This left us with 6 out of the 21 attributes, and their names and statistical values are displayed in Figures 2 and 3 above. It should be noted that these are the attributes that are sited as most effective at differentiating bacterial from viral meningitis [2,3].

The key results from the rough sets approach can be summarised by a confusion matrix, which very concisely depicts the sensitivity, specificity, positive predictive value and negative predictive value along with the total classification accuracy. In table 1 below, we depict the confusion matrices for several experiments where we employed data imputation.

Table 1. Confusion matrices of 3 rough sets based classifications using the reduced dataset (no imputation, 110 objects). The overall classification accuracy is displayed in the lower right hand corner of each subtable.

50/50	**Bacterial**	**Viral**	
Bacterial	**24**	**4**	0.857
Viral	**4**	**14**	0.778
	0.857	0.778	**0.826**
75/25			
Bacterial	**16**	**3**	0.84
Viral	**0**	**10**	1.0
	1.0	0.769	**0.896**
80/20			
Bacterial	**13**	**1**	0.929
Viral	**0**	**9**	1.0
	1.0	0.90	**0.857**

The results depicted in table 1 are of course representative of a large number of experiments that were performed (for cross validation purposes). It was interesting to note that as the size of the training set increased, the accuracy rose as well. This was a consistent trend throughout the experiments performed and is probably a reflection of the lack of data and/or consistency within the data. In order to investigate this observation more fully, we next evaluated the rule set that was generated (we use an order preserving genetic algorithm according to [9]) to sample through the reducts. First, one can examine the number of rules that are generated, their average length and the coverage. In these experiments, the number rules was fairly small – on the order of 50-400, depending on whether or not the full dataset was employed (with imputation) or the reduced dataset (i.e. complete objects – 110). It was found that with the full dataset – with imputation and hence the full complement of 21 attributes, the rule set tended to be quiet short – on the order of 40-60 rules. On the other hand, the completed dataset – without imputation (the attributes used are displayed in Figure 2) yielded a much larger number of rules – on the order of 500. In addition, the rule set for the smaller decision table (110 objects) had on average a higher cardinality and slightly less coverage than the imputed dataset (average 4 attributes versus 2). We therefore decided to corroborate the results from the rough sets approach using a PNN. The classification results from the PNN gave an accuracy of 86.3% using the reduced attributes set – with no missing items. This result was quite consistent with our results – when using the

same ration of training and testing as was employed in the rough sets experiments (see Figure1 above). We therefore concluded that the results generated from the rough sets experiment were validated not only internally using 5-fold cross validation, but also with an independent classification technique, the probabilistic neural network.

We next display a sample of the rule sets that were generated. This is one of the key features of the rough sets algorithm – the end result is a series of easily understood rules in conjunctive normal form. These rules can form the basis of an inference engine for an expert system – but at the very least lend themselves to direct comprehension by a person trained in the domain of the dataset.

Table 2. Sample rule set generated using rough sets from the same data presented in Table 1.

Lymphs([*, 26)) AND Gl([*, 150)) AND Bloodgl ([80.1,*)) => abm(0)
Lymphs([*, 26)) AND Bloodgl([*, 26)) => abm(0)

Lymphs ([26, *)) AND Gl([150.1, 260)) AND reds2([*, 13)) => abm(1)
Age([*, 4)) => abm(1)
Age([*,4)) AND Gl([150.1, 260)) => abm(1)

4. DISCUSSSION

The results presented in this paper demonstrate that rough sets is a very useful tool for the analysis of complex datasets. Even though the particular dataset employed in this study was fairly small (581 objects in total), the complexity of biomedical datasets, with all of its inherent noise and variability renders them generally resistant to full and complete analysis. In this paper, we were able to generate a classifier that has very high accuracy, reproducibility, and understandability. The accuracy generated by indicates that the system generates proper classification results on par with state-of-the art medical diagnostics. The reproducibility was evident from the cross-validation results, which indicated that the classification accuracy was robust under the given sample set. The rule set produced is in the form of a conjunctive normal form – with the attributes in question and their values stated explicitly within the rule set. The accuracy of the results were corroborated using an independent classifier – the PNN. Although the rough sets results were slightly better than the PNN algorithm. Independent corroboration is very useful and provides support to the results.

Although the results are quite promising, further work in this area is required in order to ensure that these results are broadly applicable in this domain – that we have a more substantial and complete dataset with which to work with. In this way, the resulting classifier can be enhanced and made more effective as an adjunctive diagnostic tool for various forms of meningitis.

5. REFERENCES

[1] Nigrovi, L.E., Kuppermann, N., & Malley, R. Development and validation of a Multivariate Predictive Model to Distinguish Bacterial From Aseptic Meningitis in Children in the Post-*Haemophilus influenza* Era, Pediatrics, vol 110(40, October, 2002 pp 712-719.

[2] Negrini, B., Kelleher, K.J., & Wald, E.E., Cerebrospinal Fluid Findings in Aseptic Versus Bacterial Meningitis, Pediatrics, Vol 105(2), February, 2000, pp 316-319.

[3] Spanos, A., Harrell, F.E., & Durack, D.T., Differntial Diagnosis of Acute Meningitis, JAMA, Vol 262(19), November 17, 1989, pp 2700-2707.

[4] Bonsu B.K., & Harper, M.B., Differentiating Acute Bacterial Meningitis From Acute Viral Meningitis Among Children With Cerebrospinal Fluid Pleocytosis: A Multivariable RegressionModel., Pediatric Infectious Disease Journal. 23(6):511-517, June 2004.

[5] K. Revett and A. Khan, "A Rough Sets Based Breast Cancer Decision Support System," METMBS, Las Vegas, Nevada, June, 2005

[6] Khan A & Revett.K "Data mining the PIMA Indian diabetes database using Rough Set theory with a special emphasis on rule reduction," INMIC2004, Lahore Pakistan, pp. 334-339, December, 2004

[7] Z. Pawlak . Rough Sets, International Journal of Computer and Information Sciences, 11, pp. 341-356, 1982.

[8] D. Slezak.: "Approximate Entropy Reducts". Fundamenta Informaticae, 2002.

[9] Wroblewski, J.: "Theoretical Foundations of Order-Based Genetic Algorithms". Fundamenta Informaticae 28(3-4) pp. 423–430, 1996.

[10] D. F. Specht, "Probabilistic neural networks," *Neural Networks*, vol. 3, pp. 109-118, 1990.

[11] F. Gorunescu, "Architecture of probabilistic neural networks: estimating the adjustable smoothing factor," Research Notes in Artificial Intelligence and Digital Communications, 104, pp. 56-62, 2004.