

A comparative study of some classification algorithms using WEKA and LAD algorithm

ALEXANDRA BĂICOIANU
(BRAȘOV)

ABSTRACT. In the present paper we describe and investigate the performance of novel optimization applications in classification and regression models with LAD. Also, we analyze the execution time and the accuracy of a LAD implementation and we compare the results to the appropriate classification and regression algorithms used in the machine learning literature, and their implementation in WEKA. In comparison to the commonly used machine learning algorithms implemented in the publicly available WEKA software package, the implementation of LAD was shown here to be a highly competitive classification algorithm.

KEY WORDS: dataset, classification, LAD, WEKA

1 Introduction

The possibilities of both generating and collecting data have been increasing rapidly for the last several years. Data mining is a multidisciplinary field, aiming to perform nontrivial extraction of implicit,

[◇]Alexandra Băicoianu, Faculty of Mathematics and Computer Science,
Transilvania University of Brașov, Romania,
email: a.baicoianu@unitbv.ro

previously unknown, interesting, and potentially useful information from data.

Our aim for this paper is to focus on a data mining tool, a combinatorics and optimization based data analysis method, that is able to perform classification, Logical Analysis of Data (LAD). It completely generates and analyzes a major subset of those combinations of variables which can describe the positive or negative nature of observations and it has been applied to numerous disciplines. In this paper we investigate the execution time and the accuracy of LAD. In addition, the LAD algorithm is compared with the most important classification algorithms used in the literature, and their implementation in WEKA [12]. Several detailed studies showed that the accuracy of LAD compares favorably with that of the best methods used in data analysis, with LAD usually providing results that are similar and sometimes exceed those obtained by the most frequently used methods. The value of LAD was reconfirmed and considerably strengthened in the real life medical applications. Hence, we chose for this study a certain medical dataset and an economic one. Viewing LAD in economic parameters [3] is not difficult because the performance of this methodology has been validated by many successful applications.

An important contribution of the author in this paper consists in providing and evaluating computationally the LAD models and make some comparison with some specific machine learning algorithms. We evaluate the coverage cases in the case of LAD and other specialized algorithms, as well. As an overall conclusion, we may say that LAD research has made tremendous progress since the original publications and its implementation was shown here to be a highly intelligent classification algorithm.

2 Theoretical aspects

This section provides an overview of some aspects which are relevant for understanding the main issues of this paper: LAD methodology and models, C4.5, Random Forest, SVM and MPL algorithms, WEKA package.

LAD is one of the most promising data mining methods developed to date for extracting knowledge from data. It was first introduced in the '80s in [8] and the basic algorithm was extended to the case of non-binary

data, by the binarization process, bringing the problem to the usual binary form. A recent overview of LAD can be found in [7]. LAD's practical applications to medical problems were started with the publications [2, 9]. LAD differentiates itself from other data mining algorithms by the fact that it generates and analyzes a major subset of those combinations of variables which can describe the positive or negative nature of observations, and uses optimization techniques to extract models constructed with the help of a limited number of combinatorial patterns generated in this way [6].

Many methods exist for classification in conventional data mining, mostly based on statistics: clustering, decision trees, association rules. LAD suggests a new way of analyzing data through combinatorial logic, Boolean functions, and optimization techniques. It differs from other conventional data mining methods by its ability to detect logical combinatory information about the observations. The LAD method detects patterns for which all the satisfying observations have a decidedly higher or lower show rate than the studied population. From this point of view, we can see some similarity to algorithms such as C4.5 [11], and rough sets classification [10].

In this section we briefly concentrate on the discussion and details of those algorithms that we use in the next section for the purpose of evaluating the quality of our computational results on LAD. Whenever possible we highlight some connections between particular features of these algorithms and some of the basic concepts of LAD.

C4.5 is a classification algorithm, which generates a decision tree using the training data. It uses the concept of information gain to make a tree of classificatory decisions with respect to a previously chosen target classification. Evidently, the separation of the input space into subspaces in the decision tree context is closely related to that of patterns in LAD methodology. We state the fact that in the case of decisions trees the set of patterns is significantly smaller than that of LAD. Because usually the decision tree models are simpler classifiers, they are less expensive computationally generated. LAD classifiers are more general than decision tree models. We will see in our next section that the obtained results for LAD and decision tree are close enough, even if they are quite different in some aspects.

Random forests are an ensemble learning method that operate by constructing a multitude of decision trees at training time and outputting

the class that is the mode of the classes output by individual trees. The computational cost of building trees is very small, the collective use of such simple trees has been shown to result in very accurate classification models [5]. Given that a sufficiently large number of trees is generated, we expect that random forests models have a very close performance to that of LAD models [1]. We proved these theoretical results through tests, using WEKA.

SVM is a supervised learning model that analyzes data and recognizes patterns. The typical SVM's classifier consists of a linear discriminant function that separates the training data in a very similar way to the LAD. Besides this, the optimization model for optimizing the weights of the discriminant function in LAD is almost identical to the one utilized in SVM. We will see in the next section that on the same dataset these two algorithms have tight results.

MPL is a feedforward artificial neural network model that maps sets of input data onto a set of appropriate outputs. MPL utilizes a supervised learning technique called backpropagation for training the network.

For all the next experiments, we used 3.7.11 WEKA version (<http://www.cs.waikato.ac.nz/ml/WEKA/>). WEKA contains tools for data pre-processing, classification, regression, clustering, association rules, and visualization. WEKA's main user interface is the Explorer, but essentially the same functionality can be accessed through the component-based Knowledge Flow interface and from the command line.

3 Computational Experiments

Patterns are the basic building blocks in LAD and have been shown to provide important indications about the positive or negative nature of the points covered by them. The collection of patterns used in the implementation of LAD is generated by a combinatorial enumeration process, which can be quite expensive. The number of the type of patterns generated in the process are controlled by several controls parameters. There are some studies that are concentrated on the complexity of the pattern generation process, using the concept of maximum patterns [7], we defined and used these ideas in our experiments. In the previous section, we summarily defined the LAD methodology and we described some essential classification

algorithms. Here, we discuss the comparative accuracy of the LAD models (using Ladoscope) using maximum patterns and the accuracy and results for these algorithms (using WEKA).

The first considered dataset (medicine): **breast-cancer-wisconsin.arff**. It is known from the literature that this is a clean dataset on which many data analysis methods provide highly accurate diagnostic models. We consider the classification accuracy of LAD for this dataset was 96.63%. The results were checked and reconstructed with Ladoscope by the author and they can be also found in Rutcor Research Report, [7]. What we did here was to evaluate the accuracy of some classification models using one random 10-fold cross-validation. We wanted to see how some data mining algorithms work and make obvious that LAD is a competitive method.

Using WEKA graphical user interface we can see: the number of numerical attributes (10), the number of instances (699), the name of the attributes, the minimum value, the maximum value, etc. From the Clasify menu, we chose J48 (C4.5). WEKA's result using a 10-fold cross-validation is specified in Figure 1. From the "Confusion Matrix" we can read that

```

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances      661           94.5637 %
Incorrectly Classified Instances    38           5.4363 %
Kappa statistic                    0.8799
Mean absolute error                 0.0694
Root mean squared error             0.2229
Relative absolute error             15.352 %
Root relative squared error         46.8927 %
Coverage of cases (0.95 level)     96.4235 %
Mean rel. region size (0.95 level) 53.7196 %
Total Number of Instances          699

=== Detailed Accuracy By Class ===

          TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
          0.956   0.075   0.961     0.956   0.958     0.880   0.955   0.955     2
          0.925   0.044   0.918     0.925   0.921     0.880   0.955   0.902     4
Weighted Avg.   0.946   0.064   0.946     0.946   0.946     0.880   0.955   0.937

=== Confusion Matrix ===
  a    b  <-- classified as
438  20 |  a = 2
 18 223 |  b = 4

```

Figure 1: Results for J48, DataSet: bcw.arff

661 instances were correctly classified (438+223) and 38 (the rest, out of 699) were incorrectly classified. From WEKA's menu we can change some options, like parameters, vizualization, etc.

For the same dataset, we select SMO (SVM). Exactly like J48, it has some default parameters. In practice, this algorithm takes into account a penalty term, in order to allow some observations in the training dataset to be incorrectly classified. The so-called C -parameter (default value - 1.0) dictates how much importance the model should give to the perfect separation of the training data, as opposed to the maximization of the margin of separation of the most observation. Another important parameter is the kernal function, the feature space chosen (here, polynomial function). We acquired: Correctly Classified Instances percentage 96.9957, better than J48. We can use another option that 10-fold cross validation, for example "Percentage split", we chose value 70, which means we use 70% from our data like training set and the rest like test set. For these options, we gained a percentage of 95.7143, 201 correctly classified instances, 9 incorrect, time to build model: 0.11 seconds, time to test model on training split: 0.02 seconds.

The valid options/parameters for MPL are: L , learning rate for backpropagation algorithm, default value 0.3, momentum rate for the backpropagation algorithm, default value 0.2, training time 500. This algorithm can get excellent results, but often involves (much) experimentation, taking into account the number and size of hidden layers and the value of learning rate and momentum. The results are: percentage 95.279 % (666 correctly and 33 incorrectly), time: 1.19 seconds.

In order to perform a fair comparison between LAD and these algorithms, we calibrated a few parameters of each algorithm. LAD provides a reasonable compromise between achieving superior classification performance and keeping the computational expense reasonable low. The accuracy of LAD models are at least comparable to the accuracies of the chosen algorithms.

We mention that WEKA offers the possibility to run more that one algorithm in one instance, using Experimenter Option. In Experimenter Environment we can work with multiple datasets at the time as well. We can sort the algorithm, using one comparator, Figure 2. In order to report the comparison between these classifiers, we have to know the symbols: * - significant poor, v - significant better, *blank* - none of the two.

```

Tester:   weka.experiment.PairedCorrectedTTester
Analysing: Percent_correct
Datasets: 1
Resultsets: 4
Confidence: 0.05 (two tailed)
Sorted by: Number_correct
Date:     10/26/14 7:23 PM

```

Dataset	(1) trees.J4	(2) trees	(3) funct	(4) funct
breast_cancer_92	(100) 95.01	95.89	96.75 v	95.57
	(v/ /*)	(0/1/0)	(1/0/0)	(0/1/0)

Key:

```

(1) trees.J48 '-C 0.25 -M 2' -217733168393644444
(2) trees.RandomForest '-I 10 -K 0 -S 1 -num-slots 1' 1116839470751428698
(3) functions.SMO '-C 1.0 -L 0.001 -P 1.0E-12 -N 0 -V -1 -W 1 -K \"functions.supportVector.PolyKernel -E 1.0 -C 250007\" -658588363786917.
(4) functions.MultilayerPerceptron '-L 0.3 -M 0.2 -N 500 -V 0 -S 0 -E 20 -H a' -5990607817048210779

```

Figure 2: Experimenter - Results, DataSet: bcw.arff

The second dataset (economic): **bank-data.csv**. In addition to the native ARFF data file format, WEKA has the capability to read in ".csv" format files. WEKA will recognize the attributes and during the scan of the data will compute some basic statistics on each attribute. We select J48 and we get: time : 0.01 seconds, percentage: 89.8333 % (correctly), 10.1667 % (incorrectly). LAD's results were: time: 0.07 seconds, percentage: 91.333%.

In comparison with the commonly used machine learning algorithms implemented in the publicly available WEKA software package, the implementation of LAD was shown here to be a highly competitive classification algorithm. Obviously, we can still change some parameters of the algorithms and we can still perform some tests with other algorithms from WEKA, but this is considered a future work in this area.

4 Conclusions and Future Work

In the present paper we have investigated the performance of LAD and we analyzed the execution time and its accuracy. Additionally, we briefly outlined the ideas behind several well-known classification algorithms and we have presented the validation procedures used to estimate accuracy in

our computational experiments.

The main goal of this paper was to provide and evaluate computationally the LAD models and make some comparison with some other well known algorithms of data mining. Our experiments demonstrated the utility of LAD on certain datasets. The empirical results on "bcw.arff" dataset showed that the performance of LAD is comparable with C4.5, SVM or MPL algorithms. We have acquired a very competitive classification accuracy, 96.63% correctly classified instances. In addition to producing accurate models, it gives data analysts the flexibility to set specific requirements such as statistical significance of patterns. For the "bank-data.csv" dataset, LAD gave even better results than SVM or MPL algorithms, 91.333% correctly classified instances for LAD and $\approx 89\%$ for SVM and MPL. It is part of future research plans to assess the time needed to construct the LAD models and to minimize it as much as possible, with the best results.

Moreover, we intend to develop a LAD algorithm for WEKA. As we said before, WEKA is also well-suited for developing new machine learning schemes. It can be extended to include the elementary learning schemes for research and educational purposes. If we realize this issue, then the use of LAD algorithm in classification will become easier.

Additionally, as a perspective on this presented data mining tool, we would like to extend our previous work on Haskell [4] to some pattern mining algorithms. We intend to use specific libraries, like HLearn - Haskell based library for machine learning (<https://izbicki.me/public/papers>) - in some specific problems of data mining, particularly on LAD. An interesting perspective would be to find a Haskell alternative solution for practical problems basically solved with LAD.

References

- [1] G. Alexe, P.L. Hammer. *Spanned patterns for the logical analysis of data*. Discrete Appl. Math., **154**(7) (2006), 1039–1049.
- [2] S. Alexe, et al. *Coronary Risk Prediction by Logical Analysis of Data*. Annals of Operations Research, **119** (2003), 15–42.

-
- [3] A. Băicoianu, S. Dumitrescu. *Data mining meets Economic Analysis: Opportunities and Challenges*. Bulletin of the Transilvania University of Braşov, **3**(52) (2010), series V:Economic Sciences, 185–192.
 - [4] A. Băicoianu, A. Vasilescu, et al. *Upon the performance of a Haskell parallel implementation*. Bulletin of the Transilvania University of Braşov, **6**(55) (2013) Series III: Mathematics, Informatics, Physics, 61–72.
 - [5] R.E. Banfield. *A comparison of decision tree ensemble creation techniques*. in Pattern Analysis and Machine Intelligence, IEEE Transactions on **29** (1) (2007), 173–180.
 - [6] E. Boros, P.L. Hammer, et al. *An implementation of logical analysis of data*. IEEE on Knowledge and Data Engineering, IEEE Transactions on, textbf12(2) (2000), 292–306.
 - [7] P.L. Hammer, T. Bonates. *Logical Analysis of Data: From Combinatorial Optimization to Medical Application*. Rutcort Research Report, RRR 10, 2005.
 - [8] P.L. Hammer. *The Logic of Cause-effect Relationships*. IC Multi-Attribute Decision Making OR-based Expert Systems, Germany, 1986.
 - [9] M. Lauer, S. Alexe *Use of the LAD Method for Assessing Long-Term Mortality Risk After Exercise Electrocardiography*, Circulation **106**(6) (2002), 685–690.
 - [10] Z. Pawlak. *Rough Sets: Theoretical Aspects of Reasoning about Data*. Kluwer Academic Publishers Norwell, USA, 1992.
 - [11] J.R. Quinlan. *C4.5: Programs for Machine Learning*, Morgan Kaufmann, 1993.
 - [12] I.H. Witten, E. Frank. *Data Mining: Practical Machine Learning Tools with Java Implementations*, Morgan Kaufmann, San Francisco, 1999.