Data Mining Methods Applied to a Digital Forensics Task for Supervised Machine Learning

Antonio J. Tallón-Ballesteros and José C. Riquelme

Department of Languages and Computer Systems, University of Seville
Reina Mercedes Avenue, Seville, 41012 Spain
`atallon@us.es`

Abstract. Digital forensics research includes several stages. Once we have collected the data the last goal is to obtain a model in order to predict the output with unseen data. We focus on supervised machine learning techniques. This chapter performs an experimental study on a forensics data task for multi-class classification including several types of methods such as decision trees, bayes classifiers, based on rules, artificial neural networks and based on nearest neighbors. The classifiers have been evaluated with two performance measures: accuracy and Cohen's kappa. The followed experimental design has been a 4-fold cross validation with thirty repetitions for non-deterministic algorithms in order to obtain reliable results, averaging the results from 120 runs. A statistical analysis has been conducted in order to compare each pair of algorithms by means of t-tests using both the accuracy and Cohen's kappa metrics.

Keywords: Digital forensics, Glass evidence, Data mining, Supervised machine learning, Classification model.

1 Introduction

Forensic science can be defined as the application of the science to matters of the law. A fundamental principle of forensic science is that a criminal act, or more generally a human-initiated event, produces a record of itself. The record, however imperfect, is the results of human actor(s) ant the events they set in motion producing interactions that result in changes in the environment. Object get moved or broken, marks are made, and materials are changed or transferred [1]. Forensic analysis is usually performed through experiments in lab which is expensive both in cost and time. Nowadays, data availability is increasing and the computational intelligence [2] techniques are very important in order to do automatically an accurate and fast analysis. Popescu and Farid [3] did a research about the use of statistical tools for altered photographs in the digital forensics context. Although digital forensics has been around for several decades, it is still a young science, and the body of peer-reviewed, academic literature that is essential for every science is currently relatively small, but it is growing [4]. Several kinds of evidences may be present in a forensic activity, such as fibres, paint, glass, soil, fingerprints. Depending on the types chemical tests, microscopic

techniques, molecular spectroscopy, elemental analysis, mass spectrometry, separation techniques or thermal analysis could be conducted [5].

An important area in forensic science called forensic interpretation of glass evidence is devoted to the study of several kind of glass properties (shape, structure, colour, size, thickness,…) after their breakage [6]. Forensic glass analysis tries to discriminate between several types of glasses. Moreover sometimes, a subsequent work once the glasses have been passed by an annealing process is performed. It has been applied in the case of a bi-class problem with toughened and laminated glasses [7]. Winstanley and Rydeard [8] were pioneered in talking about some annealing concepts about small glass fragments. Terry et al. [9] performed a quantitative analysis of glasses used in Australia depending on the source country.

The classification of glass fragments has been addressed with three data mining approaches in [10]. Ahmad et al. [11] worked with several samples of glass from cars or shops. The purpose of the classifier in the former case was to separate the rear, wind and side glass, and in the latter one was to distinguish heat absorbing, clear, reflective and figured floats. In the context of glass microtraces, Zadora et al. [12] proposed a quantitative elemental analysis using a scanning electron microscope with an energy dispersive X-ray spectrometer (SEM-EDX) in order to achieve a classification scheme for samples collected in Poland. Float glass samples of relevant cars in New Zealand using laser ablation inductively coupled plasma mass spectrometry (LA-ICP-MS) have been analyzed in [13]. Uzkent et al. [14] have developed a system for classifying the sound produced by the glass breaking.

2   The Problem and the Data Set

Glass may be scattered in several locations and can be produced in a wide variety of forms and compositions, and these affect the properties of this material [15]. It can occur as evidence when it is broken during the commission of a crime. Broken glass fragments ranging in size from large pieces to tiny shards may be transferred to and retained by nearby persons or objects. The mere presence of fragments of glass on the clothing of an alleged burglar in a case involving entry through a broken window may be significant evidence if fragments are found. The significance of such evidence will be enhanced if the fragments are determined to be indistinguishable in all measured properties from the broken window. On the other hand, if the recovered fragments differ in their measured properties from the glass from the broken window, then that window can be eliminated as a possible source of the glass on the subject's clothing [16].

Our digital forensics problem is to forecast the type of class on basis of the chemical analysis. The study of classification of types of glass was motivated by criminological investigation. Their data set is named Glass Identification [17], whose data come from USA Forensic Science Service. It was created by B. German and was donated by V. Spiehler to UCI (University of California, Irvine) repository [18]. Instances belong to one of the types of glass, defined in terms of their oxide content (i.e. Na, Fe, K, etc). Now, we proceed to describe the semantics of the features and the class label.

- Attribute 1. Id number: 1 to 214.
- Attribute 2. RI: refractive index.
- Attribute 3. Na: Sodium (unit measurement: weight percent in corresponding oxide, as are attributes 4-10).
- Attribute 4. Mg: Magnesium.
- Attribute 5. Al: Aluminum.
- Attribute 6. Si: Silicon.
- Attribute 7. K: Potassium.
- Attribute 8. Ca: Calcium.
- Attribute 9. Ba: Barium.
- Attribute 10. Fe: Iron.
- Class label. Type of glass. There are seven possible values:

− Building_windows_float_processed (value 1).
− Building_windows_non_float_processed (value 2).
− Vehicle_windows_float_processed (value 3).
− Vehicle_windows_non_float_processed (value 4). However, there are no instances containing this glass type.
− Containers (value 5).
− Tableware (value 6).
− Headlamps (value 7).

We have deleted the information related with the identificator of the instances and we have considered six possible output values. Table 1 summarizes the main properties of the data set taking into account the previous remarks and Table 2 depicts the values of the statistics related with glass identification features.

This problem has been used in several works. V. Spiehler experienced with a binary classification problem for the determination whether the glass was a type of "float" glass or not. She conducted a comparison test of her rule-based system, BEAGLE, the nearest-neighbor algorithm and discriminant analysis. Also, Buscema [19] has tested the Glass Identification problem in a binary form with four classifiers. Previously, Parvin et al. [20] introduced an ensemble approach and tested it with the 6-class glass problem. The multi-class version of this digital forensics task is very complex since it is difficult to classify, as literature have reported, with a high accuracy and thus this is the motivation of this chapter.

Table 1. Summary of the digital forensics problem

Patterns	Attributes	Numeric Attributes	Domain and Type	Nominal Attributes	Classes
214	9	9	Real (continuous)	0	6

Table 2. Problem statistics

Attribute	Mean	SD	Min	Max	Correlation with class
2. RI	1.5184	0.0030	1.5112	1.5339	-0.1642
3. Na	13.4079	0.8166	10.73	17.38	0.5030
4. Mg	2.6845	1.4424	0	4.49	-0.7447
5. Al	1.4449	0.4993	0.29	3.5	0.5988
6. Si	72.6509	0.7745	69.81	75.41	0.1515
7. K	0.4971	0.6522	0	6.21	-0.0100
8.Ca	8.9570	1.4232	5.43	16.19	0.0007
9. Ba	0.1750	0.4972	0	3.15	0.5751
10. Fe	0.0570	0.0974	0	0.51	-0.1879

SD standard deviation.

3   The Algorithms

Classifiers can be divided in several types [21-22]:

- Decision trees.   A possible definition of a decision tree is a simple structure based on a tree that can be used as a classifier. Each non-leaf or internal node is associated with a decision and the leaf nodes are generally associated with an outcome or class label. Each internal node tests one or more attribute values leading two or more links or branches. Each link in turn is associated with a possible value of the decision. These links are mutually distinct and collectively exhaustive. This means that it is possible to follow only one of the links and all possibilities will be taken care of—there is a link for each possibility. The interested reader is referred to Murthy's paper [23] for a deep review. We have used two representative methods like C4.5 [24] and CART [25] that stands for Classification and Regression Tree.

- Bayes classifiers. In pattern recognition, Bayes classifier [26] is popular because it is an optimal classifier. It is possible to show that the resultant classification minimizes the average probability of error. Bayes classifier is based on the assumption that information about classes in the form of prior probabilities and distributions of patterns in the class are known. It employs the posterior probabilities to assign the class label to a test pattern; a pattern is assigned the label of the class that has the maximum posterior probability. The classifier employs Bayes theorem to convert the prior probability into posterior probability based on the pattern to be classified, using the likelihood values. We have used a Bayesian network (BayesNet) which is a probabilistic graphical model that represents a set of variables and their probabilistic dependencies. Formally, Bayesian networks are directed acyclic graphs whose nodes represent variables, and whose arcs encode conditional dependencies between the variables. There are efficient algorithms that perform inference and learning in Bayesian networks [22].

- Rule-Based classifiers. Also named rule induction classifiers. The learned model is represented as a set of IF-THEN rules. Rules are a good way of representing

information or bits of knowledge [27]. In problems where classes can be characterized by general relationships, rather than just by examples (instances). It becomes attractive to build classifiers based on rules. Humans generally like explanations for most decisions. Rules, one at a time, can be directly learned from the data that is called rule induction. Each rule is a combination of conditions [22]. As an example of this classifier type we have used RIPPER (Repeated Incremental Pruning to Produce Error Reduction) [28].

- Artificial neural networks. The field of neural networks has arisen from diverse sources, ranging from the fascination of mankind with understanding and emulating the human brain, to broader issues of copying human abilities such as speech and the use of language, to the practical commercial, scientific, and engineering disciplines of pattern recognition, modeling, and prediction [29]. There are several approaches: feed-forward and recurrent neural networks [30]. We have used the feed-forward type including two well-known approaches like the Multi-Layer Perceptron (MLP) neural network [31] with a back-propagation algorithm and the Radial Basis Function (RBF) neural network [32].

- Classifiers based on nearest neighbours. One of the simplest decision procedures that can be used for classification is the nearest neighbour (NN) rule [33-34]. It classifies a sample based on the category of its nearest neighbour. When large samples are involved, it can be shown that this rule has a probability of error which is less than twice the optimum error—hence there is less than twice the probability of error compared to any other decision rule. The nearest neighbour based classifiers use some or all the patterns available in the training set to classify a test pattern. These classifiers essentially involve finding the similarity between the test pattern and every pattern in the training set. The nearest neighbour (1-NN) algorithm assigns to a test pattern the class label of its closest neighbor. We have used three 1-NN variants that differ in the distance function that compute the dissimilarity measure or distance. Euclidean, Manhattan and Chebyshev (also written as Tchebyschev) distance [35] measures have been tested in the current chapter. The resulting classifiers are called Classical 1-NN (sometimes referred as Euclidean 1-NN), Manhattan 1-NN and Chebyshev 1-NN. The first one is very common in machine learning community. Recently, Boularias and Chaib-draa [36] have compared Euclidean and Manhattan distances in the k-NN algorithm for apprenticeship learning. In other research, related with granular data modeling, Chebyshev and Euclidean distances have been used [37].

4   Experimentation

4.1   Validation Technique

The experimental design used in this chapter has been a stratified four-fold cross validation [38]. The primary idea of the four-fold cross validation procedure is to divide the full data set in four partitions of the same size; each one is used as a test set and the remaining are used as a train set. The stratification subjects to that the partitions

maintain the class distribution of the samples approximately equal as in the original data set [39]. Stochastic algorithms have been run thirty times and since we have four folds, the results are averaged by one hundred and twenty runs in order to obtain reliable results.

4.2   Performance Measures

There are several measures for assessing the models obtained by the classifiers [40]. We have gathered the following performance measures:

- Accuracy. Generally speaking, the accuracy of a classifier is the probability of correctly classifying a randomly selected instance [39]. It is also known as the number of successful hits [41]. Mathematically, the accuracy is given by:

$$Accuracy = \frac{\sum_{n=1}^{N} I(C(\boldsymbol{x_n}) = y_n)}{N} * 100 \tag{1}$$

   where $I(g)$ is a function that returns 1 if g is true and 0 otherwise, $C(\boldsymbol{x_n})$ the class label assigned to the $\boldsymbol{x_n}$ pattern by the classifier and N the total number of patterns.

- Cohen's kappa. It is an interesting alternative measure to the accuracy, since it compensates for random hits [42]. It was first introduced as a measure of agreement between observers of psychological behavior. The original intent of Cohen's kappa was to measure the degree of agreement, or disagreement, between two people observing the same phenomenon The range of Kappa values extends from positive to negative one, with positive one indicating strong agreement, negative one indicating strong disagreement, and zero indicating chance-level agreement. In order to illustrate, the calculation of Cohen's kappa from the confusion matrix we will take as a starting point a 3-class problem which confusion matrix including marginal values is shown in Table 3.

Table 3. Confusion matrix for a 3-class problem

		Predicted class			
		C1	*C2*	*C3*	Total
Correct class	*C1*	*a*	*b*	*c*	$a+b+c=C1_{corr}$
	C2	*d*	*e*	*f*	$d+e+f=C2_{corr}$
	C3	*g*	*h*	*i*	$g+h+i=C3_{corr}$
	Total	$a+d+g=C1_{pred}$	$b+e+h=C2_{pred}$	$c+f+i=C3_{pred}$	N

Being N the total number of patterns, C1, C2 and C3 the label related with class 1, 2 or 3, respectively. Their Cohen's kappa is given by:

$$Cohen's\ kappa = \frac{N*(a+e+i)-(C1_{corr}*C1_{pred}+C2_{corr}*C2_{pred}+C3_{corr}*C3_{pred})}{N^2-(C1_{corr}*C1_{pred}+C2_{corr}*C2_{pred}+C3_{corr}*C3_{pred})} \quad (2)$$

It can be generalized to the m classes

$$Cohen's\ kappa = \frac{N\sum_{i=1}^{m}CM_{ii}-\sum_{i=1}^{m}Ci_{corr}Ci_{pred}}{N^2-\sum_{i=1}^{m}Ci_{corr}Ci_{pred}} \quad (3)$$

where CM_{ii} represent the diagonal elements of the confusion matrix.

Next, we will compute both performance measures for a numeric example taken from [40] that is depicted in Table 4. The accuracy and Cohen's kappa of the confusion matrix example for the 3-class problem is as follows.

$$Accuracy = \frac{15+15+45}{100}*100 = 75\%$$

$$Cohen's\ kappa = \frac{N\sum_{i=1}^{m}CM_{ii}-\sum_{i=1}^{m}Ci_{corr}Ci_{pred}}{N^2-\sum_{i=1}^{m}Ci_{corr}Ci_{pred}} =$$

$$\frac{100\sum(15+15+45)-\sum(20*24+30*20+50*56)}{100^2-\sum(20*24+30*20+56*40)} = 0.5915$$

Table 4. Confusion matrix example for a 3-class problem

		Predicted class			
		C1	*C2*	*C3*	Total
Correct class	*C1*	15	2	3	20
	C2	7	15	8	30
	C3	2	3	45	50
	Total	24	20	56	100

Source: [40].

The Cohen's kappa value is greater than 0 (random classification) and more close to 1 (perfect classification), that indicates some classification errors. The performance is good, but it can be improved for instance by correctly classifying more samples of the class number 2.

4.3   Algorithm Implementation and Parameters

For the experimentation we have used the implementations of the algorithms described in Section 3 that are included in framework WEKA (Waikato Environment for Knowledge Analysis) version 3.7.4 [43], with the exceptions of CART and RBF taken from the version 3.5.7. We have tested the methods related with different supervised machine learning approaches such as decision trees, bayes classifiers, rule-based classifiers, artificial neural networks and classifiers based on nearest neighbours. More specifically, we have carried out experiments with the following nine algorithms: C4.5 (J48), CART (SimpleCart), BayesNet, RIPPER (JRip), MLP with a back-propagation method, RBF, Euclidean 1-NN, Manhattan 1-NN and Chebyshev 1-NN. Regarding the parameters, in the first experiment the algorithms have been run with the default values which are according to the recommendations of their own authors. In addition, these values have been used by us in some previous studies and showed a robust behavior [44]. In the second experiment we have reported the results with fined-tuned parameter values that are described in the next section.

4.4   Statistical Tests

A statistical analysis has been performed in order to find out significant differences between the results obtained by the stochastic algorithms that we have dealt with. For the non-stochastic algorithms it is not possible to carry out the analysis because we have only one result per fold and the number of freedom degrees would be low for it. Since we have one problem and several stochastic algorithms we have performed a paired t-test for comparing the algorithms two by two [45]. More specifically, we have done a two-tailed t-test at a significant level of 0.05. Let μ_1 be the mean performance of the first algorithm and μ_2 be the mean performance of the second algorithm, and $\mu_d = \mu_1 - \mu_2$, the hypotheses are the following:

— $H_0 : \mu_d = 0$. There is no difference in the mean performances of the two algorithms.
— $H_1 : \mu_d > 0$. The first algorithm seems to work better.

The t statistic is computed. For the t value we will obtain the tail area (p-value) from the t-distribution table with a number of number of freedom degrees equal to the sample size minus one of the repetitions performed by each algorithm (in our case 120-1=119). If the p-value is lower than 0.05 we reject the null hypothesis concluding that there significant differences and the first algorithm is significantly better to the second one.

Statistical analysis have been conducted for the both performances measures reported in this chapter in order to extract general conclusions about which are the most stochastic appropriate algorithms for the digital forensics problem.

5   Results

This section is structured in two subsections. The first one is devoted to report the results with the default parameter values that were proposed by their authors due they are robust in general terms; in addition we have included a statistical comparison in order to obtain an overview if there are significant differences between stochastic algorithms. The second one show the results with fined-tuned parameter values of the algorithms by means of a grid search using the training set of each fold; since trials with a different range of the parameters could conduct to other ordering of the algorithms we have not performed any kind of statistical test. In both subsections we have divided the results in two parts: one for non-stochastic algorithms and another for the stochastic ones. The accuracy and Cohen's kappa measures have been reported for each algorithm regarding to training and test phases.

5.1   Results with Default Parameter Values

Table 5 shows the results obtained with the default parameters. Taking into account the non-stochastic algorithms, the best one is Manhattan 1-NN with a test accuracy over 70% and a test Cohen's kappa very close to 0.6. The second best is Euclidean 1-NN with differences of approximately 0.5 for accuracy and 0.01 for Cohen's kappa. The next best algorithms are BayesNet, C4.5 and Chebyshev 1-NN. The best stochastic algorithms ordered by decreasing performance for both measures are CART and RIPPER. The followers are MLP and/or RBF algorithms, depending on the evaluation measure. Statistical test will let to refine these remarks by means of a two-tailed t-test for each pair of algorithms. We do not established a direct comparison between non-stochastic and stochastic algorithms due to the different number of iteration for each kind of method.

The best results published recently in the paper authored by Silva and Hruscka [46] using the same data set are similar (70% with k-NN instead of 1-NN) although there are important differences in the kind of cross validation (ten-fold versus four-fold) and that work does not contain any statistical analysis for the aforementioned problem. They have reported the mean accuracy without including neither the SD nor the Cohen's kappa measure, thus it is not possible to comment some issues about the homogeneity of the solutions or to get an overview about the global classifier performance for the different labels of the instances. Two years ago, Wang et al. [47] presented a study about the performance of extreme learning machine (ELM) and introduced a new proposal called effective ELM (EELM). Their experimental designed was performed by a hold-out getting an accuracy test (from 0 to 1) with ELM and EELM over 0.42 averaged by fifty trials.

Now, we present the statistical analysis results for the stochastic algorithms. We have done two independent kinds of tests: one for accuracy and another for Cohen's kappa that are reported in Tables 6 and 7.

Table 5. Training and test results with the accuracy and Cohen´s kappa measures for 6-class glass identification problem

Algorithm type	Classifier approach	Method	Accuracy (%)		Cohen's kappa	
			Training	Test	Training	Test
Non-stochastic	Decision Tree	C4.5	90.50±1.59	68.00±8.33	0.8700±0.0214	0.5663±0.1005
	Bayes	BayesNet	*	69.59±7.50	*	0.5830±0.0974
	Nearest neighbour	Euclidean 1-NN	100.00±0.00	*69.64±7.84*	1.0000±0.0000	*0.5867±0.1062*
	Nearest neighbour	Manhattan 1-NN	100.00±0.00	**70.13±6.85**	1.0000±0.0000	**0.5949±0.0973**
	Nearest neighbour	Chebyshev 1-NN	100.00±0.00	65.04±6.18	1.0000±0.0000	0.5222±0.0849
Stochastic	Decision Tree	CART	80.99±4.39	**67.87±2.39**	0.7358±0.0621	**0.5541±0.0343**
	Rules	RIPPER	80.63±4.65	*66.26±6.03*	0.7327±0.0648	*0.5290±0.0837*
	ANN	MLP	82.44±2.57	65.47±5.73	0.7557±0.0365	0.5180±0.0805
	ANN	RBF	79.15±2.78	65.27±8.33	0.7170±0.0383	0.5259±0.1118

Best and *second best* test results depending on the algorithm type have been highlighted in **boldface** and *italics*, respectively.

* Training results not provided by the classifier implementation.

Table 6. Statistical analysis with a two-tailed t-test for accuracy measure in the 6-class glass identification problem

Two-tailed t-test for accuracy				
Algorithm 1	Algorithm 2	p-value	t(119) statistic	Statistical test conclusion
CART	RIPPER	0.0099 *	2.6226	**CART > RIPPER**
CART	MLP	$8.868*10^{-5}$ *	4.0586	**CART > MLP**
CART	RBF	0.0023 *	3.1205	**CART > RBF**
RIPPER	MLP	0.2780	1.0899	RIPPER ≥ MLP
RIPPER	RBF	0.2200	1.2331	RIPPER ≥ RBF
MLP	RBF	0.7536	0.3146	MLP ≥ RBF

Overall accuracy ranking: $\mu_{Accuracy(CART)} > \mu_{Accuracy(RIPPER)} \geq \mu_{Accuracy(MLP)} \geq \mu_{Accuracy(RBF)}$.

  * : Significant difference at $\alpha = 0.05$.

According to the statistical test results for accuracy, we can assert that CART is the algorithm with a performance significantly better than the remaining algorithms. The second best algorithm is RIPPER but the differences with their competitors are not enough to be significant. Thus the best classifier belongs to decision tree approach and the next best to rules. Comparing the two models of neural networks, there are no significant differences although MLP is slightly better than RBF.

Table 7. Statistical analysis with a two-tailed t-test for Cohen's kappa measure in the 6-class glass identification problem

Two-tailed t-test for Cohen's kappa				
Algorithm 1	Algorithm 2	p-value	t(119) statistic	Statistical test conclusion
CART	RIPPER	0.0044 *	2.9016	**CART > RIPPER**
CART	MLP	$3.4*10^{-5}$ *	4.3051	**CART > MLP**
CART	RBF	0.0145 *	2.4813	**CART > RBF**
RIPPER	MLP	0.2827	1.0791	RIPPER $\geq$ MLP
RIPPER	RBF	0.7820	0.2774	RIPPER $\geq$ RBF
MLP	RBF	0.3678	0.9041	RBF $\geq$ MLP

Overall Cohen's kappa ranking:

$\mu_{\text{Cohen's kappa(CART)}} > \mu_{\text{Cohen's kappa(RIPPER)}} \geq \mu_{\text{Cohen's kappa(RBF)}} \geq \mu_{\text{Cohen's kappa(MLP)}}$.

$*$: Significant difference at $\alpha = 0.05$.

For Cohen's kappa, statistical test indicates that CART is significantly the best algorithm. The second best one is RIPPER that is quantitatively better than MLP and RBF. The last one neural network model is slightly better than MLP without significant differences.

5.2   Results with Fine-Tuned Parameter Values

First of all, we introduce the parameter values that we have defined for the fine setting by means of a grid search with the training set of each fold. For the 1-NN algorithm it is not possible to use specific parameters with the exception of the distance function that we have considered in the previous subsection. Table 8 presents the possible values or range of the parameters that we have selected for the fine tuning; the algorithms are sorted depending on the type, that is, first the non-stochastic ones and then the stochastic ones.

Table 9 reports the results of those algorithms obtained with the aforementioned fine-tuned parameters grouped by algorithm type and classifier approach. Also, we have included the results default with the default parameters, due to the reasons exposed at the beginning of this subsection, for the three variants of 1-NN in order to get a general view of the performance. In reference to the non-stochastic methods, the two best algorithms are C4.5 and Manhattan 1-NN, depending on the performance measure. From the stochastic aspect, the best classifier for both measures is CART, followed by RBF.

The fine setting of the parameters has shifted the performance ordering of the non-stochastic algorithms and has let to improve the results; the best classifier has now surpassed the top of 73.5% of accuracy and has reached a Cohen's kappa close to 0.595. In the context of stochastic methods, this tuning has increased the performance of the algorithms and has moved the name of the second best classifier; the best results for both measures are over 68% and 0.55 for accuracy and Cohen's kappa, respectively.

Table 8. Fine-tuned parameter values of the algorithms by means of a grid search on the training set of each fold

Algorithm	Parameter	Possible values or range	Default value	Best value
C4.5	Confidence factor (C)	{0.150, 0.175, 0.200, 0.225, 0.250}	0.25	0.175
	Minimum number of instances per leaf (M)	2-10	2	2
BayesNet	Alpha value (A) for Simple Estimator	{0.25, 0.50, 0.75}	0.50	0.75
CART	Minimal number of observations of the terminal nodes (M)	2-5	2	4
	The number of fold in the internal cross-validation	2-10	5	5
RIPPER	Folds: the amount of data used for pruning (F)	1-5	3	4
	The minimum total weight of the instances in a rule (N)	1-3	2	1
	The number of optimization runs (O)	1-3	2	3
MLP	TrainingTime: The number of epochs to train through (N)	{250, 500, 750, 1000}	500	500
	HiddenLayers: hidden layers of the neural network (H)	4-16	a = (attribs. + classes) / 2	15
RBF	NumClusters: The number of clusters for K-Means to generate (B)	1-6	2	4

Table 9. Training and test fine-tuned results with the accuracy and Cohen´s kappa measures for 6-class glass identification problem

Algorithm type	Classifier approach	Method	Accuracy (%)		Cohen's kappa	
			Training	Test	Training	Test
Non-stochastic	Decision Tree	C4.5	88.71±2.31	**73.67±2.62**	0.8511±0.0334	*0.5929±0.1006*
	Bayes	BayesNet	*	70.07±6.84	*	0.5886±0.0897
	Nearest neighbour	Euclidean 1-NN	100.00±0.00	69.64±7.84	1.0000±0.0000	0.5867±0.1062
	Nearest neighbour	Manhattan 1-NN	100.00±0.00	*70.13±6.85*	1.0000±0.0000	**0.5949±0.0973**
	Nearest neighbour	Chebyshev 1-NN	100.00±0.00	65.04±6.18	1.0000±0.0000	0.5222±0.0849
Stochastic	Decision Tree	CART	77.78±2.98	**68.22±1.93**	0.6907±0.0388	**0.5576±0.0292**
	Rules	RIPPER	82.39±4.65	66.85±4.84	0.7584±0.0640	0.5397±0.0674
	ANN	MLP	86.95±2.45	66.47±3.37	0.8200±0.0339	0.5323±0.0456
	ANN	RBF	88.28±2.35	*66.86±4.67*	0.8407±0.0320	*0.5475±0.0633*

Best and *second best* test results with fine-tuned parameters depending on the algorithm type have been highlighted in **boldface** and *italics*, respectively.

* Training results not provided by the classifier implementation.

6   Conclusions

In this chapter we have reviewed the state-of-the-art related with a digital forensics task called Glass Identification in the context of multi-class supervised learning. This problem have been tackled from some decades to the present, however the previous studies are focused on a particular issue. We have presented an empirical overview of the performance with a good number of classifiers from different machine learning approaches with two metrics like accuracy and Cohen's kappa for training and test stages, using the default parameter values in the first experiment and the fine-tuned values in the second one. We have included a statistical analysis in the first experiment that has revealed some valuable conclusions.

In the first experiment, related with the deterministic algorithms, Manhattan 1-NN obtains the best performance for accuracy and Cohen's kappa metrics. Their performance is slightly better than the Euclidean 1-NN. Our real-world problem is another sample in that nearest neighbours classifiers can be applied successfully. Thus, it has been proven that Manhattan 1-NN is better than Euclidean 1-NN, BayesNet, C4.5 and Chebyshev 1-NN. Moreover, we have reported the results of non-deterministic algorithms; however it is not possible to compare them with deterministic algorithms because the former methods have been smoothed by an average of one hundred and twenty runs versus four of the latter methods. The best non-deterministic algorithm is CART with statistically significant differences with the remaining non-deterministic methods. The second best classifier is RIPPER, however there are no significant differences with the classifiers with a lower performance. Best approaches for non-deterministic methods are, in this order, decision trees, rules and artificial neural networks. In the second experiment, the best deterministic algorithm is C4.5 or 1-NN Manhattan according to the performance evaluation measure. The best non-deterministic algorithm is CART with both measures and the second best one is the RBF neural network model.

The most important remarks taking into account both experiments are stated as follows. The fine tuning of the parameters has been very useful due to: i) From the non-stochastic algorithm perspective the best accuracy results has passed 73.5% with C4.5 classifier and are very close to 0.595 for Cohen's kappa with 1-NN Manhattan, ii) The performance of the best stochastic algorithm has reached 68.22 and 0.5576 for accuracy and Cohen's kappa, respectively. The problem tackled can be considered very difficult since, up the best of our knowledge, it is not possible, as this chapter showed, to classify the test instances with an accuracy level a 75%. A possible future research line of this chapter could try to study some pre-processing data mining techniques in order to act on the features, instances or values of the attributes.

References

1. Caddy, B.: Forensic Examination of Glass and Paint: Analysis and Interpretation. Taylor & Francis, London (2011)
2. Mumford, C.L., Jain, L.C. (eds.): Computational Intelligence. ISRL, vol. 1. Springer, Heidelberg (2009)
3. Popescu, A.C., Farid, H.: Statistical Tools for Digital Forensics. In: Fridrich, J. (ed.) IH 2004. LNCS, vol. 3200, pp. 128–147. Springer, Heidelberg (2004)
4. Kessler, G.C.: Advancing the Science of Digital Forensics. Computer 45(12), 25–27 (2012)
5. Stuart, B.H.: Forensic Analytical Techniques. John Wiley & Sons, West Sussex (2013)
6. Curran, J.M., Hicks, T.N., Buckleton, J.S.: Forensic Interpretation of Glass Evidence. CRC Press, Boca Raton (2000)
7. Newton, A.W.N., Kitto, L., Buckleton, J.S.: A study of the performance and utility of annealing in forensic glass analysis. Forensic Science International 155, 119–125 (2005)
8. Winstanley, R., Rydeard, C.: Concepts of annealing applied to small glass fragments. Forensic Science International 29, 1–10 (1985)
9. Terry, K.W., van Riessen, A., Lynch, B.F., Vowles, D.J.: Quantitative analysis of glasses used within Australia. Forensic Science International 25, 19–34 (1984)
10. Zadora, G.: Classification of Glass Fragments Based on Elemental Composition and Refractive Index. Journal of Forensic Science 54(1), 49–59 (2009)
11. Ahmad, U.K., Asmuje, N.F., Ibrahim, R., Kamaruzamanc, N.U.: Forensic Classification of Glass Employing Refractive Index Measurement. Malaysian Journal of Forensic Sciences 3(1), 1–4 (2012)
12. Zadora, G., Brozek-Mucha, Z., Parczewski, A.: A classification of glass microtraces. Problems of Forensic Sciences XLVII, 137–143 (2001)
13. Grainger, M.N.C., Manley-Harris, M., Coulson, S.: Classification and discrimination of automotive glass using LA-ICP-MS. Journal of Analytical Atomic Spectrometry 27, 1413–1422 (2012)
14. Uzkent, B., Barkana, B.D., Cevikalp, H.: Non-speech environmental sound classification using SVMs with a new set of features. International Journal of Innovative Computing, Information and Control 8(5B), 3511–3524 (2012)
15. Bottrell, M.C.: Forensic Glass Comparison: Background Information Used in Data Interpretation. Forensic Science Communications 11(2) (2009)
16. Koons, R.D., Buscaglia, J., Bottrell, M., Miller, E.T.: Forensic glass comparisons. In: Saferstein, R. (ed.) Forensic Science Handbook, 2nd edn., vol. I, pp. 161–213. Prentice Hall, Upper Saddle River (2002)
17. Evett, I.W., Spiehler, E.J.: Rule induction in forensic science. In: Knowledge Based Systems in Government, pp. 152–160. Halsted Press, London (1988)
18. Frank, A., Asuncion, A.: UCI Machine Learning Repository. University of California, School of Information and Computer Science, Irvine, CA (2010), http://archive.ics.uci.edu/ml
19. Buscema, M.: Artificial Adaptive Systems in Data Visualization: Proactive Data. In: Buscema, M., Tastle, W. (eds.) Intelligent Data Mining in Law Enforcement Analytics: New Neural Networks Applied to Real Problems, pp. 51–88 (2013)
20. Parvin, H., Minaei-Bidgoli, B., Shahpar, H.: Classifier Selection by Clustering. In: Martínez-Trinidad, J.F., Carrasco-Ochoa, J.A., Ben-Youssef Brants, C., Hancock, E.R. (eds.) MCPR 2011. LNCS, vol. 6718, pp. 60–66. Springer, Heidelberg (2011)

21. Murty, M.N., Devi, V.S.: Pattern Recognition. An Algorithmic Approach. Universities Press (India), Pvt. Ltd., London (2011)
22. Dougherty, G.: Pattern Recognition and Classification: An Introduction. Springer, New York (2013)
23. Murthy, S.K.: Automatic Construction of Decision Trees from Data: A Multi-Disciplinary Survey. Data Mining and Knowledge Discovery 2, 345–389 (1998)
24. Quinlan, J.: C4.5: Programs for Machine Learning. Morgan Kaufmann, San Francisco (1993)
25. Breiman, L., Friedman, J., Olshen, R., Stone, C.: Classification and Regression Trees. Wadsworth Int. Group, Belmont (1984)
26. Pearl, J.: Probabilistic reasoning in intelligent systems: networks of plausible inference. Morgan Kaufmann Publishers, San Francisco (1998)
27. Han, J., Kamber, M., Pei, J.: Data Mining: Concepts and Techniques. Morgan Kaufmann Publishers, Waltham (2011)
28. Cohen, W.: Fast effective rule induction. In: Proc. of the 12th Int. ICML Conf., pp. 115–123 (1995)
29. Michie, D., Spiegelhalter, D.J.: Machine Learning, Neural and Statistical Classification. Ellis Horwood, New York (1994)
30. Haykin, S.O.: Neural Networks and Learning Machines. Prentice Hall, Upper Saddle River (2009)
31. Bishop, M.: Neural Networks for Pattern Recognition. Oxford University Press, New York (1995)
32. Howlett, R.J., Jain, L.C.: Radial Basis Function Networks 1: Recent Developments in Theory and Applications. Springer, Heidelberg (2001)
33. Fix, E., Hodges, J.: Discriminatory analysis, nonparametric discrimination: consistency properties. Tech. Rep. 4, USAF School of Aviation Medicine, Randolph Field, Texas (1951)
34. Cover, T., Hart, P.: Nearest neighbor pattern classification. IEEE Transactions on Information Theory 13(1), 21–27 (1967)
35. Tan, P.N., Steinbach, M., Kumar, V.: Introduction to Data Mining. Addison-Wesley Longman Publishing Co., Boston (2005)
36. Boularias, A., Chaib-draa, B.: Apprenticeship learning with few examples. Neurocomputing 104, 83–96 (2013)
37. Bargiela, A., Pedrycz, W.: A model of granular data: a design problem with the Tchebyschev FCM. Soft Computing 9(3), 155–163 (2005)
38. Hjorth, J.S.U.: Computer intensive statistical methods: Validation model selection and bootstrap. Chapman and Hall, London (1994)
39. Kohavi, R.: A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection. In: Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI 1995), Montreal, Quebec, Canada, vol. 2, pp. 1137–1145 (1995)
40. Flach, P.: Machine Learning: The Art and Science of Algorithms that Make Sense of Data. Cambridge University Press, United Kingdom (2012)
41. Witten, I.H., Frank, E., Hall, M.A.: Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann, USA (2011)
42. Cohen, J.: A coefficient of agreement for nominal scales. Educational and Psychological Measurement 20(1), 37–46 (1960)

43. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA data mining software: an update. ACM SIGKDD Explorations Newsletter 11(1), 10–18 (2009)
44. Tallón-Ballesteros, A.J., Hervás-Martínez, C., Riquelme, J.C., Ruiz, R.: Feature selection to enhance a two-stage evolutionary algorithm in product unit neural networks for complex classification problems. Neurocomputing 114, 107–117 (2013)
45. Nisbet, R., Elder, J.F., Miner, G.: Handbook of Statistical Analysis and Data Mining Applications. Academic Press, Canada (2009)
46. Silva, J.A., Hruschka, E.R.: An experimental study on the use of nearest neighbor-based imputation algorithms for classification tasks. Data & Knowledge Engineering 84, 47–58 (2013)
47. Wang, Y., Cao, F., Yuan, Y.: A study on effectiveness of extreme learning machine. Neurocomputing 74, 2483–2490 (2011)