ARMY RESEARCH LABORATORY

Open Source Software Tools for Anomaly Detection Analysis

by Robert F. Erbacher and Robinson Pino

ARL-MR-0869 April 2014

NOTICES

Disclaimers

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

Citation of manufacturer’s or trade names does not constitute an official endorsement or approval of the use thereof.

Destroy this report when it is no longer needed. Do not return it to the originator.

Army Research Laboratory
Adelphi, MD 20783-1197

ARL-MR-0869 **April 2014**

Open Source Software Tools for Anomaly Detection Analysis

Robert F. Erbacher and Robinson Pino
Computational and Information Sciences Directorate, ARL

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. **REPORT DATE** *(DD-MM-YYYY)* April 2014	**2. REPORT TYPE** Final	**3. DATES COVERED (From - To)** September 2013

4. TITLE AND SUBTITLE
Open Source Software Tools for Anomaly Detection Analysis

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Robert F. Erbacher and Robinson Pino

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
U.S. Army Research Laboratory
ATTN: RDRL-CIN-D
2800 Powder Mill Road
Adelphi, MD 20783-1197

8. PERFORMING ORGANIZATION REPORT NUMBER
ARL-MR-0869

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited.

13. SUPPLEMENTARY NOTES
POC email: renee.e.etoty.civ@mail.mil

14. ABSTRACT

The goal of this report is to perform an analysis of software tools that could be employed to perform basic research and development of Anomaly-Based Intrusion Detection Systems. The software tools reviewed include; Environment for Developing KDD-Applications Supported by Index-Structures (ELKI), RapidMiner, SHOGUN (toolbox) Waikato Environment for Knowledge Analysis (Weka) (machine learning), and Scikit-learn. From the analysis, it is recommended to employ the SHOGUN (toolbox) or Scikit-learn as both tools are written in C++ and offers an interface for Python. The python language software is currently employed as a research tool within our in-house team of researchers.

15. SUBJECT TERMS
anomaly detection, survey, data mining

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
					Renee E. Etoty
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**			**19b. TELEPHONE NUMBER** (*Include area code*)
Unclassified	Unclassified	Unclassified	UU	22	(301) 394-1835

Contents

List of Figures

List of Tables

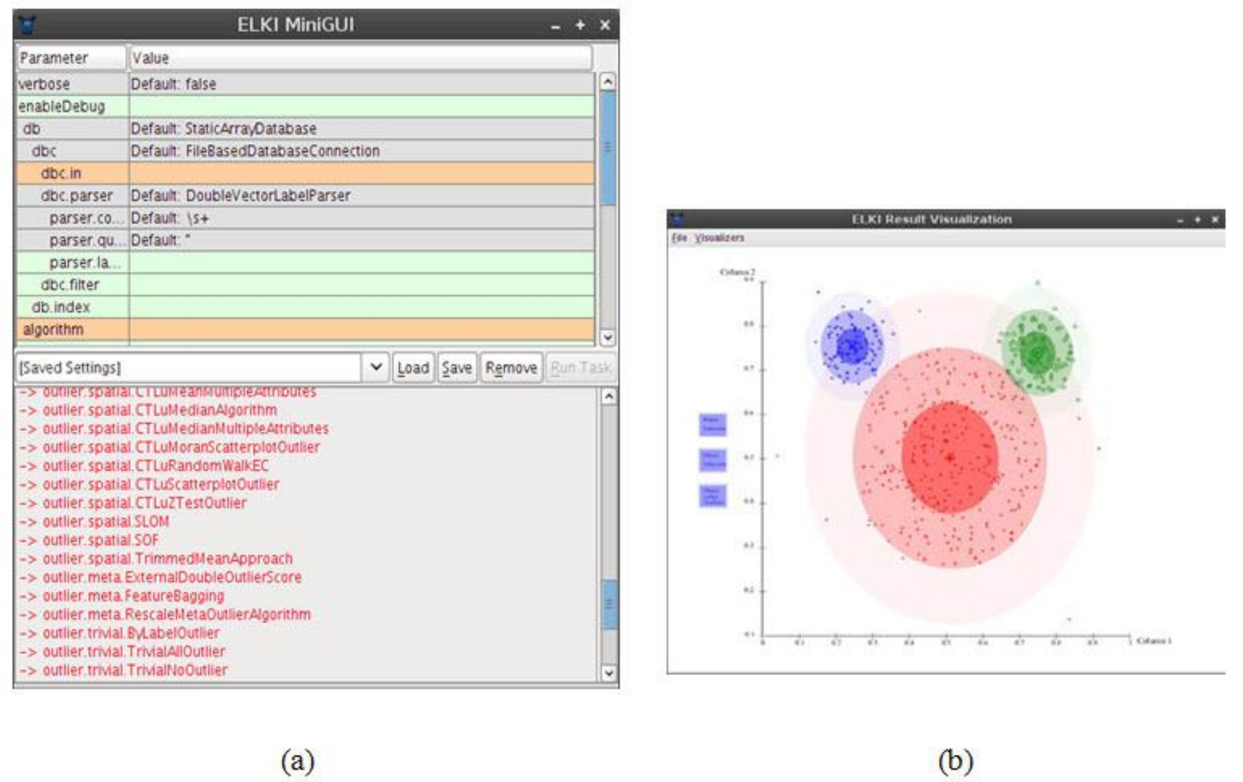
1. Introduction

Anomaly-based intrusion detection is the concept for detecting computer intrusions and misuse by monitoring network and computer activity and classifying it as either normal or anomalous (*1*). Classification is commonly based on heuristics or rules, rather than patterns or signatures, and will detect any type of misuse that falls out of normal system operation (*2*). In the case of signature based detection, the system can only detect attacks for which a signature has previously been created. In order to determine what attack traffic is, the system must be taught to recognize normal system activity. This can be accomplished using artificial intelligence techniques or neural networks (*1*). Another method is to define what normal usage of the system comprises using a strict mathematical model, and flag any deviation from this as an attack, known as strict anomaly detection (*1*). The goal of this report is to determine the suitability of current open source software packages in their usage and ability to enable our in-house team of researchers to perform basic research on anomaly-based intrusion detection algorithms.

2. Environment for Developing KDD-Applications Supported by Index Structures (ELKI)

The software tool ELKI stands for Environment for Developing KDD-Applications Supported by Index Structures and is a knowledge discovery in databases (KDD), data mining, and software framework developed for use in research and teaching by the database systems research unit of Professor Hans-Peter Kriegel at the Ludwig Maximilian University of Munich, Germany (*3*). The ELKI software package is written in Java[*] and intended to allow development and a platform for independent evaluation of data mining algorithms (*4*). The software framework is open source for scientific usage; see figure 1 for an overview.

[*] Java is a registered trademark of Oracle.



Parameter	Value
verbose	Default: false
enableDebug	
db	Default: StaticArrayDatabase
dbc	Default: FileBasedDatabaseConnection
dbc.in	
dbc.parser	Default: DoubleVectorLabelParser
parser.co...	Default: \s+
parser.qu...	Default: "
parser.la...	
dbc.filter	
db.index	
algorithm	

```
-> outlier.spatial.CTLuMeanMultipleAttributes
-> outlier.spatial.CTLuMedianAlgorithm
-> outlier.spatial.CTLuMedianMultipleAttributes
-> outlier.spatial.CTLuMoranScatterplotOutlier
-> outlier.spatial.CTLuRandomWalkEC
-> outlier.spatial.CTLuScatterplotOutlier
-> outlier.spatial.CTLuZTestOutlier
-> outlier.spatial.SLOM
-> outlier.spatial.SOF
-> outlier.spatial.TrimmedMeanApproach
-> outlier.meta.ExternalDoubleOutlierScore
-> outlier.meta.FeatureBagging
-> outlier.meta.RescaleMetaOutlierAlgorithm
-> outlier.trivial.ByLabelOutlier
-> outlier.trivial.TrivialAllOutlier
-> outlier.trivial.TrivialNoOutlier
```

Figure 1. ELKI (a) user interface and (b) output results (*4*).

ELKI provides a suit of algorithms that include: K-means clustering, anomaly detection, spatial index structures, apriori algorithm, dynamic time warping, and principal component analysis. However, an internet search for publications using this particular software application platform yields results authored by the software developers. In 2011, a book chapter published by Achert et al. (*5*) talks about spatial outlier detection: data, algorithms, and visualization. The manuscript focuses on showcasing ELKI's ability to integrate a geographic/geospatial information system (GIS) and a data mining system (DMS) within a single framework supported by the tool. In the demonstration, the authors demonstrated an integrated GIS-DMS system for performing advanced data mining tasks such as outlier detection on geospatial data, but which also allows the interaction with an existing GIS (*5*).

3. RapidMiner

The RapidMiner software, formerly Yet Another Learning Environment (YALE), is an environment for machine learning, data mining, text mining, predictive analytics, and business analytics. It is used for research, education, training, rapid prototyping, application development, and industrial applications (*6*). The software is distributed under the Affero General Public License (AGPL) open source license and has been hosted by SourceForge since 2004 (*7*).

RapidMiner provides data mining and machine learning procedures including: data loading and transformation (extract, transform, load [ETL]), data preprocessing and visualization, modeling, evaluation, and deployment. Two examples of graphical results are shown in figure 2. The data mining processes can be made up of arbitrarily nestable operators, described in eXtensible Markup Language (XML) files and created in RapidMiner's graphical user interface (GUI). RapidMiner is written in the Java programming language, and can be used for text mining, multimedia mining, feature engineering, data stream mining and tracking drifting concepts, development of ensemble methods, and distributed data mining (*7*). In addition, advanced features can be purchased as a commercial version of the base software and are available at the rapid-i.com Web site. In particular, beyond the free community edition, three enterprise software packages can be purchased from small, standard, and developer editions, which offer an increasing number of options and capabilities, respectively. The algorithms included in the RapidMiner software include: machine learning, data mining, text mining, predictive analytics, and business analytics.

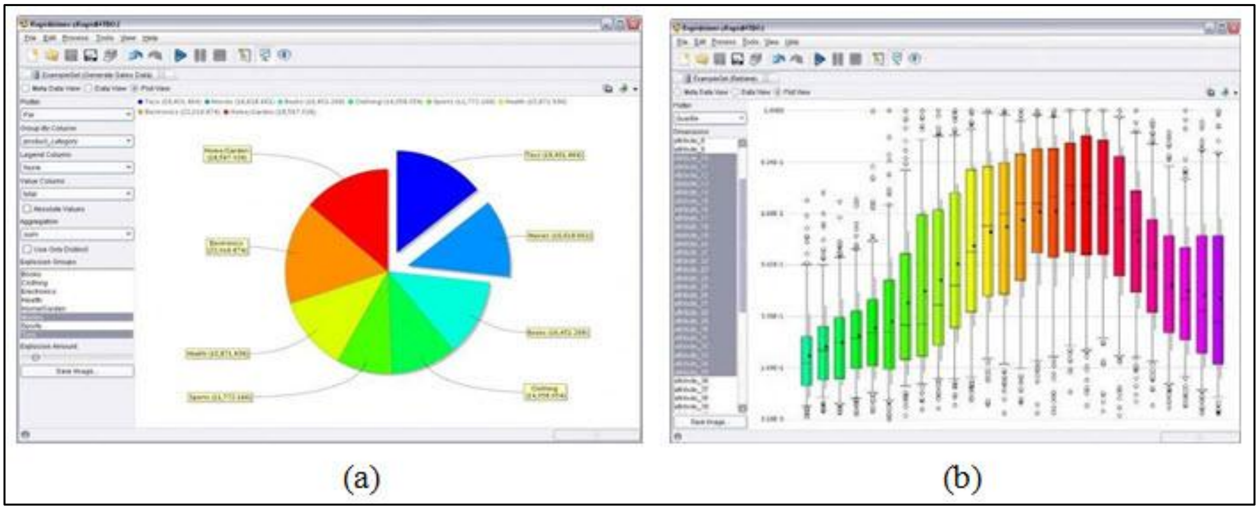


Figure 2. RapidMiner output results (*7*).

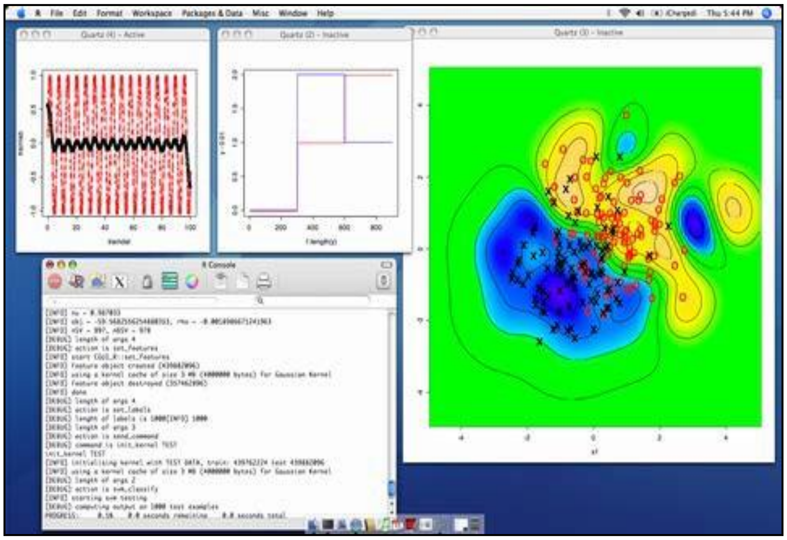
4. SHOGUN (toolbox)

The focus of SHOGUN is on kernel machines such as support vector machines for regression and classification problems. SHOGUN also offers a full implementation of Hidden Markov models. The core of SHOGUN is written in C++ and offers interfaces for MATLAB,* Octave, Python,† R, Java, Lua, Ruby, and C#. SHOGUN has been under active development since 1999. Today there is a user community all over the world using SHOGUN as a base for research and

*MATLAB is a registered trademark of The MathWorks, Inc.

†Python is a registered trademark of Python Software Foundation.

education, and contributing to the core package. SHOGUN is a free software, open source toolbox written in C++. It offers numerous algorithms and data structures for machine learning problems. SHOGUN is licensed under the terms of the GNU General Public License version 3 or later (*8*). Figure 3 shows a screenshot of SHOGUN's code and output results. The software can be obtained from the official Web site (*9*).



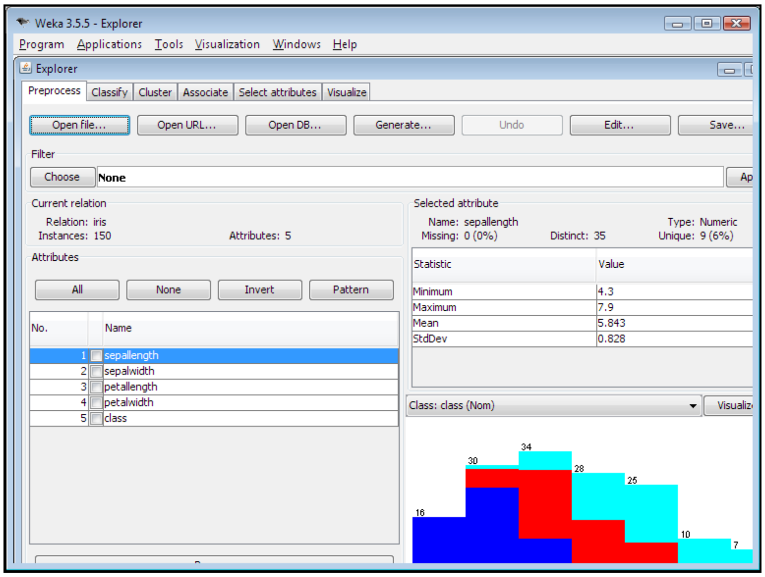
```
[INFO] nu = 0.567853
[INFO] obj = -59.5682556254408753, rho = -0.0054566671241963
[INFO] nSV = 997, nBSV = 978
[DEBUG] length of args 4
[DEBUG] action is set_features
[INFO] start CGUI_R::set_features
[INFO] Feature object created (439662096)
[INFO] using a kernel cache of size 3 MB (4000000 bytes) for Gaussian Kernel
[INFO] Feature object destroyed (357462096)
[INFO] done
[DEBUG] length of args 4
[DEBUG] action is set_labels
[DEBUG] length of labels is 1000[INFO] 1000
[DEBUG] length of args 3
[DEBUG] action is send_command
[DEBUG] command is init_kernel TEST
init_kernel TEST
[INFO] initialising kernel with TEST DATA, train: 439762224 test 439862096
[INFO] using a kernel cache of size 3 MB (4000000 bytes) for Gaussian Kernel
[DEBUG] length of args 2
[DEBUG] action is svm_classify
[INFO] starting svm testing
[DEBUG] computing output on 1000 test examples
PROGRESS:     0.10    0.0 seconds remaining    0.0 seconds total
```

Figure 3. Screenshot of SHOGUN software tool results (*8*).

Among the software tool packages reviewed, SHOGUN offers the most features for research and development. Some of SHOGUN's algorithms and features included in the software tool are: support vector machines, dimensionality reduction, online learning, clustering, and implemented kernels for numeric data analysis algorithms. Over 20 publications about SHOGUN are featured on its Wikipedia page (*8*).

5. Waikato Environment for Knowledge Analysis (Weka) (Machine Learning)

Weka is a suite of machine learning software written in Java, developed at the University of Waikato, New Zealand. The Weka workbench contains a collection of visualization tools and algorithms for data analysis and predictive modeling, together with GUIs for easy access to this functionality. Weka is free software available under the GNU General Public License (*10*). The Weka software is available for download at the official Web site (*11*), see figure 4.



Filter: None

Relation: iris
Instances: 150
Attributes: 5

Name: sepallength
Type: Numeric
Missing: 0 (0%)
Distinct: 35
Unique: 9 (6%)

Statistic	Value
Minimum	4.3
Maximum	7.9
Mean	5.843
StdDev	0.828

No.	Name
1	sepallength
2	sepalwidth
3	petallength
4	petalwidth
5	class

Class: class (Nom)

Figure 4. Screenshot of Weka software tool results (*10*).

Some of the features of this software tool include: data preprocessing, clustering, expectation maximization, classification, regression, visualization, and feature selection. The primary learning methods in Weka are "classifiers," and they induce a rule set or decision tree that models the data. Weka also includes algorithms for learning association rules and clustering data. All implementations have a uniform command-line interface. A common evaluation module measures the relative performance of several learning algorithms over a given data set (*12*).

6. Scikit-Learn

Scikit-learn (formerly scikits.learn) is an open source machine learning library for the Python programming language (*13*). It features various classifier engines, regression, and clustering algorithms including support vector machines, logistic regression, naive Bayes, k-means, and DBSCAN; and is designed to interoperate with NumPy and SciPy (*13*). The scikit license is open source, commercially usable, and the software code can be downloaded on the official Web site (*14*).

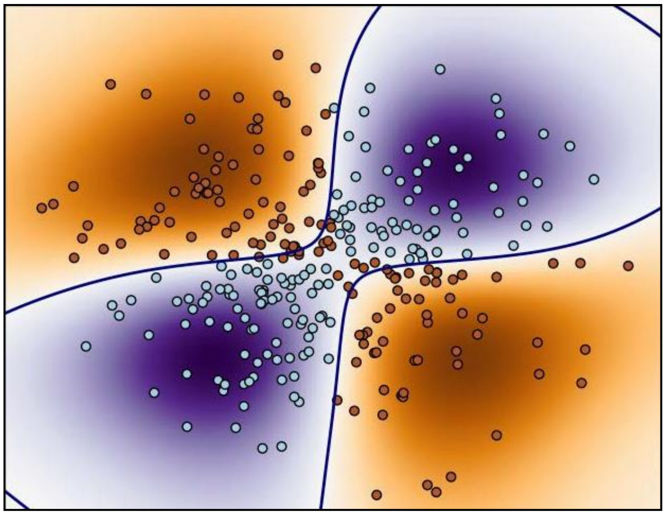


Figure 5. Sckit-learn results performing binary classification using nonlinear Support Vector Classification (SVC) with Radial Basis Function (RBF) kernel. The target to predict is an Exclusive Or (XOR) of the inputs. The color map illustrates the decision function learned by the SVC (*14*).

Scikit-learn is a Python module integrating a wide range of machine learning algorithms for supervised and unsupervised problems. The software package focuses on bringing machine learning to nonspecialists using a general-purpose high-level language; it has minimal dependencies, and is distributed under the simplified Berkeley Software Distribution (BSD) license, for use in both academic and commercial settings (*15*).

7. Results and Discussion

We have reviewed several open source software tools for performing research and development on anomaly detection for network security. After reviewing the flexibility and popularity of usage, we believe that in order to proceed with our evaluation we should select two packages for additional in-house testing and evaluation. Table 1 describes the two anomaly detection tools that we feel offer the most flexibility and the most number of anomaly detection algorithms for our in-house research purposes. From the table, we can see that the two packages share most of the basic algorithms that we can use in-house for performing basic research in anomaly detection. Therefore, we have submitted a formal request within our branch to install SHOGUN and Scikit-learn within the U.S. Army Research Laboratory's (ARL) computers and we are awaiting feedback.

Table 1. Side-by-side comparison of algorithms offered by SHOGUN and Scikit-learn.

Shogun	Scikit-Learn
Support vector machines Dimensionality reduction algorithms: PCA, Kernel PCA, Locally Linear Embedding, Hessian Locally Linear Embedding, Local Tangent Space Alignment, Linear Local Tangent Space Alignment, Kernel Locally Linear Embedding, Kernel Local Tangent Space Alignment, Multidimensional Scaling, Isomap, Diffusion Maps, Laplacian Eigenmaps Online learning algorithms: such as SGD-QN, Vowpal Wabbit Clustering algorithms: k-means and GMM Kernel Ridge Regression Support Vector Regression Hidden Markov Models K-Nearest Neighbors Linear discriminant analysis Kernel Perceptrons Kernels for numeric data: linear gaussian polynomial sigmoid kernels The supported kernels for special data include: Spectrum Weighted Degree Weighted Degree with Shifts	Supervised learning: Generalized Linear Models Support Vector Machines Stochastic Gradient Descent Nearest Neighbors Gaussian Processes Partial Least Squares Naive Bayes Decision Trees Ensemble methods Multiclass and multilabel algorithms Feature selection Semi-Supervised Linear and Quadratic Discriminant Analysis Unsupervised learning: Gaussian mixture models Manifold learning Clustering Decomposing signals in components (matrix factorization problems) Covariance estimation Novelty and Outlier Detection Hidden Markov Models

8. Conclusions

In this report, we have performed a review of various software tools that can be leveraged in-house to perform basic research and development of anomaly-based intrusion detection algorithms. Out of the five software tools described, it is recommended to employ the Scikit or

SHOGUN (toolbox) as both tools are written in C++ and offer an interface for Python. The python language software is commonly employed as a research tool within our in-house team of researchers.

9. References

1. Anomaly-Based Intrusion Detection System, Wikipedia, http://en.wikipedia.org/wiki/Anomaly-based_intrusion_detection_system (accessed September 1, 2012).

2. Wang, K.; Stolfo, S. J. *Anomalous Payload-Based Network Intrusion Detection. Recent Advances in Intrusion Detection*. Springer Berlin 2004, pp. 203.

3. ELKI, Wikipedia, http://en.wikipedia.org/wiki/Environment_for_DeveLoping_KDD-Applications_Supported_by_Index-Structures (accessed September 1, 2012).

4. ELKI: Environment for Developing KDD-Applications Supported by Index-Structures, http://elki.dbs.ifi.lmu.de/ (accessed September 1, 2012).

5. Echtert, E.; Hettab, A.; Kriegel, H. -P.; Schubert, E.; Zimek, A. Spatial Outlier Detection: Data, Algorithms, Visualizations. *Lecture Notes in Computer Science, Advances in Spatial and Temporal Databases* **2011**, *6849*, pp. 512–516.

6. RapidMiner, Wikipedia, http://en.wikipedia.org/wiki/RapidMiner (accessed September 1, 2012).

7. RapidMiner – Data Mining, ETL, OLAP, BI. Sourceforge. Geeknet, Inc. http://sourceforge.net/projects/rapidminer/ (accessed July 4, 2012).

8. SHOGUN (toolbox), Wikipedia, http://en.wikipedia.org/wiki/SHOGUN_%28toolbox%29 (accessed September 1, 2012).

9. SHOGUN – A Large Scale Machine Learning Toolbox, http://www.SHOGUN-toolbox.org/ (accessed September 1, 2012).

10. Weka (machine learning), Wikipedia, http://en.wikipedia.org/wiki/Weka_%28machine_learning%29 (accessed September 1, 2012).

11. Weka, The University of Waikato, http://www.cs.waikato.ac.nz/~ml/weka/ (accessed September 1, 2012).

12. Witten, I. H.; Frank, E.; Trigg, L.; Hall, M.; Holmes, G.; Cunningham, S. J. Weka: Practical Machine Learning Tools and Techniques with Java Implementations, Emerging Knowledge Engineering and Connectionist-Based Information Systems: Proceedings of the ICONIP/ANZIIS/ANNES'99 International Workshop "Future Directions for Intelligent Systems and Information Sciences," University of Otago, Dunedin, New Zealand, 22–23 November 1999, pp. 192–196.

13. Scikit-learn, Wikipedia, http://en.wikipedia.org/wiki/Scikit-learn (accessed September 1, 2012).

14. Scikit-learn, Machine Learning in Python, http://scikit-learn.org/stable/ (accessed September 1, 2012).

15. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Duchesnay, E. (2011) Scikit-Learn: Machine Learning in Python. *The Journal of Machine Learning Research*, *12*, pp. 2825–2830.

Appendix – List of Algorithm Per Tool

ELKI

Cluster analysis:

K-means clustering

Expectation-maximization algorithm

Single-linkage clustering

DBSCAN (Density-Based Spatial Clustering of Applications with Noise)

OPTICS (Ordering Points To Identify the Clustering Structure), including the extensions

OPTICS-OF, DeLi-Clu, HiSC, HiCO and DiSH

SUBCLU (Density-Connected Subspace Clustering for High-Dimensional Data)

Anomaly detection:

LOF (Local outlier factor) OPTICS-OF

DB-Outlier (Distance-Based Outliers) LOCI (Local Correlation Integral)

LDOF (Local Distance-Based Outlier Factor) EM-Outlier

Spatial index structures:

R-tree R*-tree M-tree

Evaluation:

Receiver operating characteristic (ROC curve) Scatter plot

Histogram

Parallel coordinates

Other:

Apriori algorithm Dynamic time warping Principal component analysis

RapidMiner

Machine learning

This appendix appears in its original form, without editorial change.

Data mining Text mining Predictive analytics Business analytics.

SHOGUN

Support vector machines

Dimensionality reduction algorithms:

PCA, Kernel PCA, Locally Linear Embedding, Hessian Locally Linear Embedding, Local Tangent Space Alignment, Linear Local Tangent Space Alignment, Kernel Locally Linear Embedding, Kernel Local Tangent Space Alignment, Multidimensional Scaling, Isomap, Diffusion Maps, Laplacian Eigenmaps

Online learning algorithms:

such as SGD-QN, Vowpal Wabbit

Clustering algorithms:

k-means and GMM Kernel Ridge Regression Support Vector Regression Hidden Markov Models

K-Nearest Neighbors

Linear discriminant analysis

Kernel Perceptrons

Implemented kernels for numeric data include:

linear gaussian polynomial sigmoid kernels

The supported kernels for special data include:

Spectrum

Weighted Degree

Weighted Degree with Shifts

Weka

Data mining:

Data preprocessing

Clustering

Expectation maximization

Classification Regression Visualization

Feature selection

Scikit

Supervised learning: Generalized Linear Models Support Vector Machines Stochastic Gradient Descent Nearest Neighbors

Gaussian Processes Partial Least Squares Naive Bayes

Decision Trees

Ensemble methods

Multiclass and multilabel algorithms

Feature selection

Semi-Supervised

Linear and Quadratic Discriminant Analysis

Unsupervised learning: Gaussian mixture models Manifold learning Clustering

Decomposing signals in components (matrix factorization problems) Covariance estimation

Novelty and Outlier Detection

Hidden Markov Models

INTENTIONALLY LEFT BLANK.

List of Symbols, Abbreviations, and Acronyms

AGPL	Affero General Public License
ARL	U.S. Army Research Laboratory
BSD	Berkeley Software Distribution
DMS	data mining system
ELKI	Environment for Developing KDD-Applications Supported by Index-Structures
ETL	extract, transform, load
GIS	geographic/geospatial information system
GUI	graphical user interface
KDD	knowledge discovery in databases
RBF	Radial Basis Function
SVC	Support Vector Classification
Weka	Waikato Environment for Knowledge Analysis
XML	eXtensible Markup Language
XOR	Exclusive Or
YALE	Yet Another Learning Environment

No. of Copies	**Organization**
1 (PDF)	DEFENSE TECHNICAL INFORMATION CTR DTIC OCA
2 (PDF)	DIRECTOR US ARMY RSRCH LAB RDRL CIO LL RDRL IMAL HRA RECORDS MGMT
1 (PDF)	GOVT PRINTG OFC A MALHOTRA
2 (PDF)	DIR USARL RDRL CIN D R ERBACHER R PINO