KEEL Data-Mining Software Tool: Data Set Repository, Integration of Algorithms and Experimental Analysis Framework

鍾岳峰

元智大學

pchomekimojuf@gmail.com

March 18, 2015

Overview

Outline for section 1

INTRODUCTION:

- KEEL pays special attention to the implementation of evolutionary learning and soft computing based techniques for **Data Mining** problems including regression, classification, clustering, pattern mining and so on.
- The aim of this paper is to present **three new aspects of KEEL**: **KEEL-dataset**, a data set repository which includes the data set partitions in the KEEL format and shows some results of algorithms in these data sets; **some guidelines for including new algorithms in KEEL**, helping the researchers to make their methods easily accessible to other authors and to compare the results of many approaches already included within the KEEL software; and **a module of statistical procedures developed** in order to provide to the researcher a suitable tool to contrast the results ob- tained in any experimental study.

INTRODUCTION:

- **Data Mining (DM)** is the process for automatic discovery of high level knowledge by obtaining information from real world, large and complex data sets [26], and is the core step of a broader process, called **Knowledge Discovery from Databases (KDD)**.
- **Evolutionary Algorithms (EAs)** [14] are optimization algorithms based on natural evolution and genetic processes.
- They are currently considered to be one of **the most successful search techniques** for complex problems in **Artificial Intelligence**.
- They have proven to be an important technique both for **learning and knowledge extraction**, making them a promising technique in DM [8, 16, 22, 24, 35, 46].

INTRODUCTION:

- In the last few years, **many DM software tools** have been developed.
- Only **a few are available** as open source software.
- **Open source tools** can play an important role as is pointed out in [39].
- KEEL (Knowledge Extraction based on Evolutionary Learning) [5] is a **open source Java software tool** which empowers the user to assess the behavior of **evolutionary learning and Soft Computing** based techniques for different kinds of DM problems: **regression, classification, clustering, pattern mining and so on**.

Outline for section 2

KEEL DESCRIPTION:

The version of KEEL presently available consists of the following function blocks (see Fig. 1):

KEEL DESCRIPTION:

function blocks

- Data Management
- Design of Experiments
- Educational Experiments

KEEL DESCRIPTION:

main features of KEEL

- It presents **a large collection of EAs** for predicting models,**pre-processing and post-processing**.
  It also contains some state-of-the-art methods for **different areas of DM** such as **decision trees, fuzzy rule based systems or interval rule-based learning**.
- It has **a statistical library** to analyze results of algorthms.
- Some algorithms have been developed using **Java Class Library for Evolutionary Computation (JCLEC)** [43].
- The software is aimed at **creating experiments containing multiple data sets** and algorithms connected among themselves to **obtain an expected results**.
- KEEL also allows **the creation of experiments in on-line mode**, aiming to provide an educational support in order to **learn the operation of the algorithm** included.

Outline for section 3

KEEL-DATASET:

In this section we present the KEEL-dataset repository.

- A detailed categorization of the considered data sets and a description of their characteristics. **Tables for the data sets in each category have been also created.**
- A descriptions of the papers which have used the partitions of data sets available in the KEEL-dataset repository. These descriptions include **results tables, the algorithms used and additional material**.

KEEL-DATASET: Data sets webpages

The categories of the data sets have been derived from the topics addressed in the experimental studies.

Data sets

Classification

- Standard data sets
- Imbalanced data sets
- Multi instance data sets
- Data sets with missing values

Regression

- Regression data sets

Unsupervised (Clustering and Associations)

- Unsupervised data sets

Low quality

- Low quality data sets

Experimental studies and results with these data sets

Classification

- Experimental studies in supervised classification
- Experimental studies with imbalanced data sets
- Experimental studies with multi instance data sets

Regression

- Experimental studies in regression

Unsupervised (Clustering and Associations)

- Experimental studies in unsupervised learning

Low quality

- Experimental studies with low quality data

FIGURE 2: KEEL-dataset webpage (`http://keel.es/datasets.php`)

KEEL-DATASET: Data sets webpages

The categories in which the data sets are divided are the following:

1. Classification problems.
    - Standard data sets.
    - Imbalanced data sets [6, 28, 41].
    - Multi instance data sets [12].
    - Data sets with missing values.
2. Regression problems.
3. Unsupervised (Clustering and Associations) problems.
4. Low quality data [37].

KEEL-DATASET: Experimental study webpages

Introduction

This section shows some relevant research papers in which some of the classification data sets avalaible in KEEL-dataset have been employed.

For each study, we provide its reference (plain text and BibTeX formats), abstract and summary. A pdf version the article can also be downloaded. Additionally, we offer complementary material about the experimental studies carried up: Algorithms tested, data sets employed and results obtained (XLS and CVS formats).

Experimental studies and results with these data sets

Jump to year: 2010 **(1)**

Year 2010 (1):

A. Fernandez, S. García, J. Luengo, E. Bernadó-Mansilla, F. Herrera, Genetics-Based Machine Learning for Rule Induction: State of the Art, Taxonomy and Comparative Study. IEEE Transactions on Evolutionary Computation, in press (2010).

Link to: Data sets, algorithms and experimental results. **Link to:** Website associated to this paper.

FIGURE 4: Keel-dataset experimental studies with standard classification data sets webpage

KEEL-DATASET: Experimental study webpages

Each paper can contain up to four links:

These webpages contains **published journal publications** which use the correspondent kind of data sets in the repository.

- The first link is the PDF file of the paper.
- The second link is the Bibtex reference of the paper.
- At the bottom on the left link Data sets, algorithms and experimental results is always present. It references to the particular Keel-dataset webpage for such paper.
- At the bottom on the right link Website associated to this paper is only present for some papers which have a particular and external webpage related with them.

Moreover, the results are detailed and listed in CSV and XLS (Excel) formatted files.

Outline for section 4

INTEGRATION OF NEW ALGORITHMS INTO THE KEEL TOOL: Introduction to the KEEL codification features

- The KEEL philosophy tries to include the fewest possible constraints for the developer, in order to ease the inclusion of new algorithms within this tool.
- We enumerate the list of details to take into account before codifying a method for the KEEL software, which is also detailed at the KEEL Reference Manual (http://www.keel.es/documents/KeelReferenceManualV1.0.pdf).

INTEGRATION OF NEW ALGORITHMS INTO THE KEEL TOOL: Introduction to the KEEL codification features

- The programming language used is Java.
- In KEEL,every method uses a configuration file to extract the values of the parameters which will be employed during its execution.

Each configuration file has the following structure:

- algorithm: Name of the method.
- inputData: A list with the input data files of the method.
- outputData: A list with the output data files of the method.
- parameters: A list of parameters of the method, containing the name of each parameter and its value (one line is employed for each one).

INTEGRATION OF NEW ALGORITHMS INTO THE KEEL TOOL: Introduction to the KEEL codification features

```
algorithm = Genetic Algorithm
inputData = ``../datasets/iris/iris.dat'' ...
outputData = ``../results/iris/result0.tra'' ...


Seed = 12345678
Number of Generations = 1000
Crossover Probability = 0.9
Mutation Probability = 0.1
...
```

INTEGRATION OF NEW ALGORITHMS INTO THE KEEL TOOL: Introduction to the KEEL codification features

- The input data-sets follow a specific format that **extends the "arff" files** by completing **the header with more metadata information** about the attributes of the problem.
- The output format consists of a header,which **follows the same scheme** as the input data, and two columns with the output values for each example separated by a whitespace.

INTEGRATION OF NEW ALGORITHMS INTO THE KEEL TOOL: Introduction to the KEEL codification features

- Although the list of constraints is short, the KEEL development team have created a simple template that **manages all these features**.

Our KEEL template includes four classes

- **Main**: This class contains the main instructions for launching the algorithm.
- **ParseParameters**: This class manages all the parameters, from the input and output files, to every single parameter stored in the parameters file.
- **myDataset**: This class is an interface between the classes of the API data-set and the algorithm.
- **Algorithm**: This class is devoted to storing the main variables of the algorithm and to naming the different procedures for the learning stage.

INTEGRATION OF NEW ALGORITHMS INTO THE KEEL TOOL: Introduction to the KEEL codification features

- The template can be downloaded following the link **http://www.keel.es/software/KEEL_template.zip**, which additionally supplies the user with the whole API data-set together with the classes for managing files and the random number generator.

INTEGRATION OF NEW ALGORITHMS INTO THE KEEL TOOL: Encoding example using the "Steady-State Genetic Algorithm for Extracting Fuzzy Classification Rules From Data" method

- We will show **how this template enables the programming within KEEL** to be straightforward, since the **user does not need to pay attention to the specific KEEL constraints** because they are completely covered by the functions **implemented in the template.**
- To illustrate this, we have selected one classical and simple method, the SGERD procedure [33].

INTEGRATION OF NEW ALGORITHMS INTO THE KEEL TOOL: Encoding example using the “Steady-State Genetic Algorithm for Extracting Fuzzy Classification Rules From Data” method

- Neither the **Main** nor the **ParseParameters** classes need to be modified, and we just need **to focus** our attention on **the Algorithm class** and the inclusion of **two new functions in myDataset**.
- We enumerate below the steps for adapting this class to this specific algorithm:

INTEGRATION OF NEW ALGORITHMS INTO THE KEEL TOOL: Encoding example using the "Steady-State Genetic Algorithm for Extracting Fuzzy Classification Rules From Data" method

1.

- we must store all the parameters values with in the constructor of the algorithm. Each parameter is selected with the **getParameter** function using its corresponding position in the parameter file, whereas the optional output files are obtained using the function **getOutputFile**.
- Furthermore, the constructor must check the capabilities of the algorithm, related to the data-set features, that is, whether it has missing values, real or nominal attributes, and so on.

INTEGRATION OF NEW ALGORITHMS INTO THE KEEL TOOL: Encoding example using the "Steady-State Genetic Algorithm for Extracting Fuzzy Classification Rules From Data" method

2.

- we execute the main process of the algorithm(procedure execute).
- If everything is alright, we perform the algorithm's operations.
- In the case of the SGERD method we must first build the Data Base (DB) and then generate an initial Rule Base (RB).
- Next, the GA is executed in order to find the best rules in the system.

INTEGRATION OF NEW ALGORITHMS INTO THE KEEL TOOL: Encoding example using the "Steady-State Genetic Algorithm for Extracting Fuzzy Classification Rules From Data" method

3.

- We write in an output file the DB and the RB to save the generated fuzzy model, and then we continue with the classification step for both the validation and test files.
- The **doOutput** procedure simply iterates all examples and returns the predicted class as a string value (in regression problems it will return a double value).
- This prediction is carried out in the **classificationOutput** function, which only runs the Fuzzy Reasoning Method of the generated RB (noted in boldface)

INTEGRATION OF NEW ALGORITHMS INTO THE KEEL TOOL: Encoding example using the "Steady-State Genetic Algorithm for Extracting Fuzzy Classification Rules From Data" method

4.

- Finally, we show the new functions that are implemented in the **myDataset** class in order to obtain some necessary information from the training data during the rule learning stage.
- We must point out that the remaining functions of this class remain unaltered.

INTEGRATION OF NEW ALGORITHMS INTO THE KEEL TOOL: Encoding example using the "Steady-State Genetic Algorithm for Extracting Fuzzy Classification Rules From Data" method

- Once the algorithm has been implemented, it can be executed directly on a terminal with the parameters file as an argument.
- Nevertheless, when included within the KEEL software, the user can create a complete experiment with automatically generated scripts for a batch-mode execution.
- Furthermore, we must clarify that the "validation file" is used when an instance- selection preprocessing step is performed, and contains the original training set data; hence, the training and validation files match up in the remaining cases.

INTEGRATION OF NEW ALGORITHMS INTO THE KEEL TOOL: Encoding example using the "Steady-State Genetic Algorithm for Extracting Fuzzy Classification Rules From Data" method

- Finally, we should point out that the complete source code for the SGERD method (together with the needed classes for the fuzzy rule generation step) can be downloaded at **http://www.keel.es/software/SGERD_source. zip**.

Outline for section 5

STATISTICAL TOOLS AND EXPERIMENTAL STUDY:

- One of the important features of the KEEL software tool is the availability of a complete package of statistical procedures, developed with the aim of providing to **the researcher a suitable tool to contrast the results obtained in any experimental study performed** inside the KEEL environment.

STATISTICAL TOOLS AND EXPERIMENTAL STUDY: KEEL Statistical Tests

- Nowadays, the use of statistical tests **to improve the evaluation process of the performance of a new method** has become a widespread technique in the field of Data Mining [10, 19, 20].
- Usually, they are employed inside the framework of any experimental analysis **to decide** when an algorithm is better than other one.
- This task, which may not be trivial, has become necessary to confirm when a new proposed method **offers a significant improvement over the existing methods** for a given problem.

STATISTICAL TOOLS AND EXPERIMENTAL STUDY: KEEL Statistical Tests

- There exist two kinds of test: **parametric** and **non-parametric**, depending of the concrete type of data employed.
- As a general rule, a non-parametric test **is less restrictive than** a parametric one, although it is less robust than a parametric when data are well conditioned.
- Parametric tests have been commonly used in the analysis of experiments in DM.
- Nonparametric tests can be employed in the analysis of experiments, providing to the researcher a practical tool to use when the previous assumptions can not be satisfied.

- Table 1 shows the procedures existing in the KEEL statistical package. For each test, a reference and a brief description is given (an extended description can be found in the Statistical Inference in Computational Intelligence and Data Mining website and in the KEEL website).

STATISTICAL TOOLS AND EXPERIMENTAL STUDY: KEEL Statistical Tests

Procedure	Ref.	Description
5x2cv-f test	[11]	Approximate f statistical test for 5x2 cross validation
T test	[9]	Statistical test based on the Student's t distribution
F test	[25]	Statistical test based on the Snedecor's F distribution
Shapiro-Wilk test	[40]	Variance test for normality
Mann-Whitney U test	[27]	U statistical test of difference of means
Wilcoxon test	[44]	Nonparametric pairwise statistical test
Friedman test	[17]	Nonparametric multiple comparisons statistical test
Iman-Davenport test	[31]	Derivation from the Friedman's statistic (less conservative)
Bonferroni-Dunn test	[38]	Post-Hoc procedure similar to Dunnet's test for ANOVA
Holm test	[30]	Post-Hoc sequential procedure (most significant first)
Hochberg test	[29]	Post-Hoc sequential procedure (less significant first)
Nemenyi test	[34]	Comparison with all possible pairs
Hommel test	[7]	Comparison with all possible pairs (less conservative)

TABLE 1: Statistical procedures available in KEEL

- In this section, we present a case study as an example of the functionality and process of creating an experiment with the KEEL software tool.
- This experimental study is focused on the comparison between the new algorithm imported (SGERD) and several evolutionary rule-based algorithms, and employs a set of supervised classification domains available in KEEL-dataset.
- Several statistical procedures available in the KEEL software tool will be employed to contrast the results obtained.

STATISTICAL TOOLS AND EXPERIMENTAL STUDY: Case of study

Algorithms and classification problems

- **Five representative evolutionary rule learning methods** have been selected to carry out the experimental study: Ant-Miner, CO-Evolutionary Rule Extractor (CORE), HIerarchical DEcision Rules (HIDER), Steady-State Genetic Algorithm for Extracting Fuzzy Classification Rules From Data (SGERD) and **Tree Analysis** with Randomly Generated and Evolved Trees (TARGET) methodology.

STATISTICAL TOOLS AND EXPERIMENTAL STUDY: Case of study

Method	Ref.	Description
Ant-Miner	[36]	An Ant Colony System based using a heuristic function based in the entropy measure for each attribute-value
CORE	[42]	A coevolutionary method which employs as fitness measure a combination of the true positive rate and the false positive rate
HIDER	[2, 4]	A method which iteratively creates rules that cover randomly selected examples of the training set
SGERD	[33]	A steady-state GA which generates a prespecified number of rules per class following a GCCL approach
TARGET	[23]	A GA where each chromosome represents a complete decision tree.

TABLE 2: Algorithms tested in the experimental study

STATISTICAL TOOLS AND EXPERIMENTAL STUDY: Case of study

Name	#Ats	#Ins	#Cla	Name	#Ats	#Ins	#Cla
Haberman	3	306	2	Wisconsin	9	699	2
Iris	4	150	3	Tic-tac-toe	9	958	2
Balance	4	625	3	Wine	13	178	3
New Thyroid	5	215	3	Cleveland	13	303	5
Mammographic	5	961	2	Housevotes	16	435	2
Bupa	6	345	2	Lymphography	18	148	4
Monk-2	6	432	2	Vehicle	18	846	4
Car	6	1728	4	Bands	19	539	2
Ecoli	7	336	8	German	20	1000	2
Led-7	7	500	10	Automobile	25	205	6
Pima	8	768	2	Dermatology	34	366	6
Glass	9	214	7	Sonar	60	208	2

TABLE 3: Data sets employed in the experimental study

Setting up the Experiment under KEEL software

- The graph in Figure 6 represents the flow of data and results from the algorithms and statistical techniques.
- A node can represent an initial data flow (group of data sets), a pre-process/post-process algorithm, a learning method, test or a visualization of results module.
- They can be distinguished easily by the color of the node.
- All their parameters can be adjusted by clicking twice on the node.
- Notice that KEEL incorporates the option of configuring the number of runs for each probabilistic algorithm, including this option in the configuration dialog of each node (3 in this case study).

STATISTICAL TOOLS AND EXPERIMENTAL STUDY: Case of study

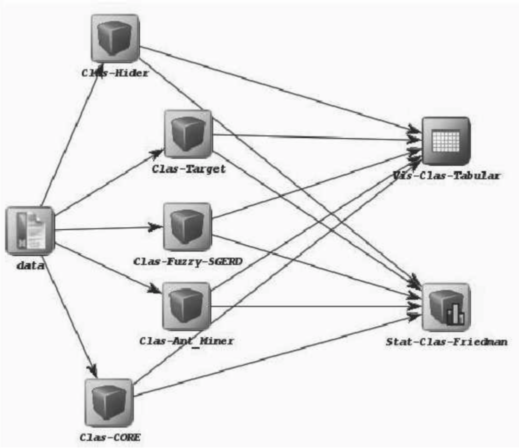

FIGURE 6: Graphical representation of the experiment in KEEL

- Table 4 shows the parameter's values selected for the algorithms employed in this experiment (they have been taken from their respective papers following the indications given by the authors).

Algorithm	Parameters
Ant-Miner	Number of ants: 3000, Maximum uncovered samples: 10, Maximum samples by rule: 10
	Maximum iterations without converge: 10
CORE	Population size: 100, Co-population size: 50, Generation limit: 100
	Number of co-populations: 15, Crossover rate: 1.0
	Mutation probability: 0.1, Regeneration probability: 0.5
HIDER	Population size: 100, Number of generations: 100, Mutation probability: 0.5
	Cross percent: 80, Extreme mutation probability: 0.05, Prune examples factor: 0.05
	Penalty factor: 1, Error coefficient: 1
SGERD	Number of Q rules per class: Computed heuristically, Rule evaluation criteria = 2
TARGET	Probability of splitting a node: 0.5, Number of total generations for the GA: 100
	Number of trees generated by crossover: 30, Number of trees generated by mutation: 10
	Number of trees generated by clonation: 5, Number of trees generated by immigration: 5

TABLE 4: Parameter' values employed in the experimental study

Results and Analysis

- This subsection describes and discusses the results obtained from the previous experiment configuration.
- Tables 5 and 6 show the results obtained in training and test stages, respectively.
- For each data set, the average and standard deviations in accuracy obtained by the module Vis-Clas-Tabular are shown, with the best results stressed in **boldface**.

STATISTICAL TOOLS AND EXPERIMENTAL STUDY: Case of study

Data set	Ant Miner Mean	Ant Miner SD	CORE Mean	CORE SD	HIDER Mean	HIDER SD	SGERD Mean	SGERD SD	TARGET Mean	TARGET SD
Haberman	**79.55**	1.80	76.32	1.01	76.58	1.21	74.29	0.81	74.57	1.01
Iris	97.26	0.74	95.48	1.42	**97.48**	0.36	97.33	0.36	93.50	2.42
Balance	73.65	3.38	68.64	2.57	75.86	0.40	76.96	2.27	**77.29**	1.57
New Thyroid	**99.17**	0.58	92.66	1.19	95.97	0.83	90.23	0.87	88.05	2.19
Mammographic	81.03	1.13	79.04	0.65	**83.60**	0.75	74.40	1.43	79.91	0.65
Bupa	**80.38**	3.25	61.93	0.89	73.37	2.70	59.13	0.68	68.86	0.89
Monk-2	97.22	0.30	87.72	7.90	97.22	0.30	80.56	0.45	**97.98**	7.90
Car	77.95	1.82	**79.22**	1.29	70.02	0.02	67.19	0.08	77.82	0.29
Ecoli	87.90	1.27	67.03	3.69	**88.59**	1.77	73.02	0.86	66.22	4.69
Led7Digit	59.42	1.37	28.76	2.55	**77.64**	0.42	40.22	5.88	34.24	3.55
Pima	71.86	2.84	72.66	2.62	**77.82**	1.16	73.71	0.40	73.42	2.62
Glass	81.48	6.59	54.26	1.90	**90.09**	1.64	53.84	2.96	45.07	0.90
Wisconsin	92.58	1.65	94.71	0.64	**97.30**	0.31	93.00	0.85	96.13	0.64
Tic-tac-toe	69.62	2.21	69.46	1.20	69.94	0.53	69.94	0.53	**69.96**	2.20
Wine	**99.69**	0.58	99.06	0.42	97.19	0.98	91.76	1.31	85.19	1.58
Cleveland	60.25	1.35	56.30	1.97	**82.04**	1.75	46.62	2.23	55.79	2.97
Housevotes	94.28	1.84	**96.98**	0.43	**96.98**	0.43	**96.98**	0.43	**96.98**	0.43
Lymphography	77.11	5.07	65.99	5.43	**83.70**	2.52	77.48	3.55	75.84	4.43
Vehicle	59.52	3.37	36.49	3.52	**84.21**	1.71	51.47	1.19	51.64	2.52
Bands	67.61	3.21	66.71	2.01	**87.13**	2.15	63.84	0.74	71.14	2.01
German	71.14	1.19	70.60	0.63	**73.54**	0.58	67.07	0.81	70.00	1.37
Automobile	69.03	8.21	31.42	7.12	**96.58**	0.64	52.56	1.67	45.66	6.12
Dermatology	86.18	5.69	31.01	0.19	**94.91**	1.40	72.69	1.04	66.24	1.81
Sonar	74.68	0.79	53.37	0.18	**98.29**	0.40	75.69	1.47	76.87	1.18
Average	79.52	2.51	68.16	2.14	**86.09**	1.04	71.76	1.37	72.43	2.33

TABLE 5: Average results and standard deviations of training accuracy obtained

STATISTICAL TOOLS AND EXPERIMENTAL STUDY: Case of study

Data set	Ant Miner		CORE		HIDER		SGERD		TARGET	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
Haberman	72.55	5.27	72.87	4.16	**75.15**	4.45	74.16	2.48	71.50	2.52
Iris	96.00	3.27	92.67	4.67	**96.67**	3.33	**96.67**	3.33	92.93	4.33
Balance	70.24	6.21	70.08	7.11	69.60	3.77	75.19	6.27	**75.62**	7.27
New Thyroid	**90.76**	6.85	**90.76**	5.00	90.28	7.30	88.44	6.83	86.79	5.83
Mammographic	81.48	7.38	77.33	3.55	**82.30**	6.50	74.11	5.11	79.65	2.11
Bupa	57.25	7.71	61.97	4.77	65.83	10.04	57.89	3.41	**65.97**	1.41
Monk-2	**97.27**	2.65	88.32	8.60	**97.27**	2.65	80.65	4.15	96.79	5.15
Car	77.26	2.59	**79.40**	3.04	70.02	0.16	67.19	0.70	77.71	2.70
Ecoli	58.58	9.13	64.58	4.28	**75.88**	6.33	72.08	7.29	65.49	4.29
Led7Digit	55.32	4.13	27.40	4.00	**68.20**	3.28	40.00	6.75	32.64	6.75
Pima	66.28	4.26	73.06	6.03	73.18	6.19	**73.71**	3.61	73.02	6.61
Glass	53.74	12.92	45.74	9.36	**64.35**	12.20	48.33	5.37	44.11	5.37
Wisconsin	90.41	2.56	92.38	2.31	**96.05**	2.76	92.71	3.82	95.75	0.82
Tic-tac-toe	64.61	5.63	**70.35**	3.77	69.93	4.73	69.93	4.73	69.50	2.73
Wine	92.06	6.37	**94.87**	4.79	82.61	6.25	87.09	6.57	82.24	7.57
Cleveland	**57.45**	5.19	53.59	7.06	55.86	5.52	44.15	4.84	52.99	1.84
Housevotes	93.56	3.69	**97.02**	3.59	**97.02**	3.59	**97.02**	3.59	96.99	0.59
Lym	73.06	10.98	65.07	15.38	72.45	10.70	72.96	13.59	**75.17**	10.59
Vehicle	53.07	4.60	36.41	3.37	**63.12**	4.48	51.19	4.85	49.81	5.85
Bands	59.18	6.58	64.23	4.23	62.15	8.51	62.71	4.17	**67.32**	6.17
German	66.90	3.96	69.30	1.55	**70.40**	4.29	66.70	1.49	70.00	0.49
Automobile	53.74	7.79	32.91	6.10	**62.59**	13.84	50.67	10.27	42.82	13.27
Dermatology	81.16	7.78	31.03	1.78	**87.45**	3.26	69.52	4.25	66.15	4.25
Sonar	71.28	5.67	53.38	1.62	52.90	2.37	73.45	7.34	**74.56**	8.34
Average	72.22	5.97	66.86	5.01	**75.05**	5.69	70.27	5.20	71.06	4.87

TABLE 6: Average results and standard deviations of test accuracy obtained

method is the one with the highest power.

- Focusing on the test results, the average accuracy obtained by **Hider** is the highest one.
- However, this estimator does not reflect whether or not the differences among the methods are significant.
- For this reason, we have carried out an statistical analysis based on multiple comparison procedures (see Appendix B for a full description), by including a node called Stat-Clas- Friedman in the KEEL experiment.

STATISTICAL TOOLS AND EXPERIMENTAL STUDY: Case of study

Algorithm	Ranking
AntMiner	3.125
CORE	3.396
Hider	**2.188**
SGERD	3.125
Target	3.167

TABLE 7: Average Rankings of the algorithms by Friedman procedure

Friedman Value	p-value	Iman-Davenport Value	p-value
8.408	0.0777	2.208	0.0742

TABLE 8: Results of the Friedman and Iman-Davenport Tests

i	Algorithm	Unadjusted p	p_{Holm}	p_{Hoch}
1	CORE	0.00811	0.032452	0.03245
2	Target	0.03193	0.09580	0.03998
3	AntMiner	0.03998	0.09580	0.03998
4	SGERD	0.03998	0.09580	0.03998

TABLE 9: Adjusted p-values. Hider is the control algorithm

Outline for section 6

CONCLUDING REMARKS:

- In this case, the results obtained have been contrasted through a statistical analysis following the indications given in [18], concluding that the Hider method is the best performing method when compared with the remaining methods analyzed in this study.

CONCLUDING REMARKS:

The objective of this paper was to present three new aspects of KEEL:

- KEEL-dataset, a data set repository that includes the data set partitions in the KEEL format and shows some results obtained in these data sets.
- Some basic guidelines that the developer may take into account to facilitate the implementation and integration of new approaches within the KEEL software tool.
- A module of statistical procedures which let researchers contrast the results obtained in any experimental study using statistical tests. This task, which may not be trivial, has become necessary to confirm when a new proposed method offers a significant **improvement over the existing methods** for a given problem.

Thank you for your attention