Corrales, D. C., Ledezma, A. & Corrales, J. C. (2020, mayo). A case-based reasoning system for recommendation of data cleaning algorithms in classification and regression tasks. Applied Soft Computing, 90, 106180.

DOI: 10.1016/j.asoc.2020.106180

Applied Soft Computing Journal xxx (xxxx) xxx

Contents lists available at ScienceDirect

Applied Soft Computing Journal

journal homepage: www.elsevier.com/locate/asoc

A case-based reasoning system for recommendation of data cleaning algorithms in classification and regression tasks

David Camilo Corrales [a,b,*], Agapito Ledezma [a], Juan Carlos Corrales [b]

[a] Departamento de Informática, Universidad Carlos III de Madrid, 28911 Leganes, Madrid, Spain
[b] Grupo de Ingeniería Telemática, Universidad del Cauca, Sector Tulcán, Popayán, Colombia

ARTICLE INFO

Article history:
Received 17 May 2019
Received in revised form 7 February 2020
Accepted 12 February 2020
Available online xxxx

Keywords:
Case-based reasoning
Classification
Regression

ABSTRACT

Recently, advances in Information Technologies (social networks, mobile applications, Internet of Things, etc.) generate a deluge of digital data; but to convert these data into useful information for business decisions is a growing challenge. Exploiting the massive amount of data through knowledge discovery (KD) process includes identifying valid, novel, potentially useful and understandable patterns from a huge volume of data. However, to prepare the data is a non-trivial refinement task that requires technical expertise in methods and algorithms for data cleaning. Consequently, the use of a suitable data analysis technique is a headache for inexpert users. To address these problems, we propose a case-based reasoning system (CBR) to recommend data cleaning algorithms for classification and regression tasks. In our approach, we represent the problem space by the meta-features of the dataset, its attributes, and the target variable. The solution space contains the algorithms of data cleaning used for each dataset. We represent the cases through a Data Cleaning Ontology. The case retrieval mechanism is composed of a filter and similarity phases. In the first phase, we defined two filter approaches based on clustering and quartile analysis. These filters retrieve a reduced number of relevant cases. The second phase computes a ranking of the retrieved cases by filter approaches, and it scores a similarity between a new case and the retrieved cases. The retrieval mechanism proposed was evaluated through a set of judges. The panel of judges scores the similarity between a query case against all cases of the case-base (ground truth). The results of the retrieval mechanism reach an average precision on judges ranking of 94.5% in top 3 (P@3), for top 7 (P@7) 84.55%, while in top 10 (P@10) 78.35%.

1. Introduction

The digital information era is an inevitable trend. Recently, advances in Information Technologies (social networks, mobile applications, Internet of Things, etc.) generate a deluge of digital data [1–3]. The massive amount of data is exploited by knowledge discovery (KD) process, which identifies valid, novel, potentially useful and understandable patterns from a huge volume of data [4,5]. Several knowledge discovery tools simplify data analysis and management. According to Gartner 2018 Magic Quadrant for Data Science and Machine Learning Platforms, KNIME [6], RapidMiner [7], SAS [8], Alteryx [9] and H20.ai [10] are the leader tools for knowledge discovery.

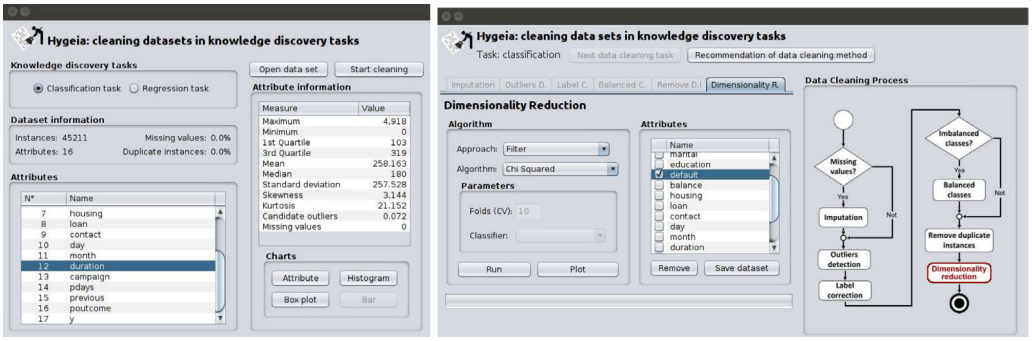
These KD tools provide different techniques, and they facilitate the gathering, application, inspection, and evaluation of data analysis and their results. However, these KD tools lack guidance as to which techniques can or should be used in which contexts [11].

Consequently, the use of a suitable data analysis technique becomes a headache for inexpert users. They are uncertain about methods to be confidently used and often resort to trial and error [11]. This problem occurs mainly in the data preparation phase. It is commonly known that 50%–80% of data analysis time is spent on pre-processing, also several data cleaning algorithms are available and their performance can vary considerably [12]. Additionally, the formulation of precise guidelines for the recommendation of data cleaning techniques is often difficult or even impossible [13,14]. Thus, the specialists rely on years of accumulated tacit experience, although, sometimes the experience is hard to express explicitly [15].

Reuse of past experiences is a powerful and frequently applied way to solve common problems. Case-based reasoning is a paradigm that uses knowledge acquired from past experiences (also named cases) to solve a given problem [11,16,17]. The CBR

* Corresponding author at: Departamento de Informática, Universidad Carlos III de Madrid, 28911 Leganes, Madrid, Spain.
E-mail addresses: dcorrales@unicauca.edu.co, davidcamilo.corrales@alumnos.uc3m.es (D.C. Corrales), ledezma@inf.uc3m.es (A. Ledezma), jcorral@unicauca.edu.co (J.C. Corrales).

https://doi.org/10.1016/j.asoc.2020.106180
1568-4946/



(a) Statistics of dataset loaded (b) Data cleaning process

Fig. 1. Hygeia: conceptual framework for data cleaning in knowledge discovery tasks. Fig. 1a contains information of data quality issues found in dataset. Tabs in Fig. 1b correspond to data cleaning tasks; data cleaning process is depicted in the right side.
Source: [1]

cycle is divided into four main steps (4R), retrieve the most similar case, reuse the knowledge of the retrieved case to solve the problem, revise the proposed solution and retain the solution with aim to solve similar cases in the future [18,19]. Several works [20–30] proposed recommendation of learners through case-based reasoning systems, however these CBR works are not focused on recommending data cleaning algorithms for classification or regression tasks.

In order to address the problem stated, we proposed a case-based reasoning (CBR) system to recommend the suitable data cleaning algorithms for classification and regression tasks. The rest of the paper is organized as follows. Section 2 discusses the related works. Section 3 presents the proposed CBR. Section 4 provides the CBR results and Section 5 presents the conclusions and future works.

2. Background

In this section, we present related works about case-based reasoning systems for knowledge discovery tasks. In addition, we explain our previous works in order to highlight the contribution of proposed CBR.

2.1. Case-based reasoning systems for knowledge discovery tasks

From knowledge discovery tasks, several authors recommend data mining algorithms through case-based reasoning systems. The authors of [20,21] built a plug-in for IBM SPSS Modeler named CITRUS. Cases are represented by data mining workflows modeled in IBM SPSS. Based on data mining task descriptions, CITRUS loads the most similar case through hierarchical planner, which builds partial workflows from data mining operators.

In [22] authors proposed an Algorithm Selection Tool (AST) to support the selection of classification and regression models. The case-base contains 80 cases composed of dataset meta-features. Also, AST defines filters based on user preferences, whether the produced model is interpretable true or false and training speed and testing time, fast or slow.

The MiningMart project [25] aims at the reuse of successful preprocessing practices (discretization, handling of null values, aggregation of attributes into a new one, collection of sequences from time-stamped data) of data stored in SQL databases. Cases are described through an ontology with informal annotations, as the goals and constraints of each problem.

Works presented in [26–29] propose an Ontology to store the expert rules of a CBR expressed in SWRL. Authors represent cases by a dataset of meta-features as number of examples, attributes and classes, mean kurtosis, mean skewness, etc. K-nearest neighbor and arithmetic similarity functions were used as a retrieval mechanism. The CBR system returns two scores: one based on similarity and the other one based on user satisfaction. After a case has been selected, the proposed system guides the user through practices of five phases of CRISP-DM methodology (business understanding, data preparation, modeling, and evaluation). A similar approach [23] uses data mining ontologies combined with the CRISP-DM methodology. It also uses the rules stored in ontologies. Unfortunately, there are several missing details about this approach.

Authors of [24] developed a data mining assistant to select a classification model. The retrieval mechanism is based on k-nearest neighbor. Unfortunately, this work lacks details on the approach.

In [30] a CBR for data preparation in electronic diabetes records was built. Experts in this work include the handling of missing values, feature selection, feature weighing, outlier detection, and normalization. The pre-processing is performed sequentially on the raw case base data (60 cases) to produce a new high-quality case base. At retrieval phase, authors use K-nearest neighbor algorithm with the local–global approach.

Previous works are directly related to our proposal (recommendation of data mining algorithms), however, the CBR for knowledge discovery tasks are not focused on recommending the suitable data cleaning algorithms for classification or regression tasks. In Section 3 we propose a case-based reasoning to recommend the suitable data cleaning methods for classification and regression tasks.

2.2. Contribution of case-based reasoning to previous works

In this sub-section, we explain the contribution by proposed CBR to previous works. In the past, we built a conceptual framework and an ontology for analysis of data quality issues in classification and regression tasks.

The conceptual framework (named Hygeia) provides a guidance to address data quality issues [1,31] as missing values, outliers, mislabeled classes, imbalanced classes, duplicate instances and high dimensionality. In order to prepare the datasets, the conceptual framework follows a sequence of data cleaning tasks (imputation, outliers detection, label correction, balanced classes, remove duplicate instances and dimensionality reduction) as shown Fig. 1b. Each data cleaning tasks shows to the users its respective approaches and methods (i.e., dimensionality reduction lists the methods of three approaches: filter, wrapper and embedded). The conceptual framework was developed in NetBeans IDE 8.2 using Swing API forms.

Please cite this article as: D.C. Corrales, A. Ledezma and J.C. Corrales, A case-based reasoning system for recommendation of data cleaning algorithms in classification and regression tasks, Applied Soft Computing Journal (2020) 106180, https://doi.org/10.1016/j.asoc.2020.106180.

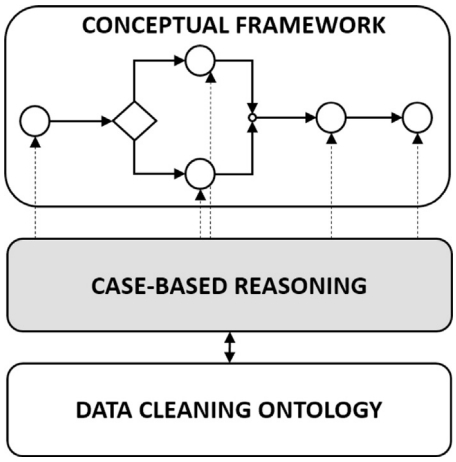


Fig. 2. CBR contribution to previous works.

In the other hand, data the cleaning ontology represents the knowledge of data quality issues and data cleaning algorithms for classification and regression tasks [31]. The ontology provides information to conceptual framework of data quality issues, data cleaning approaches, and configuration parameters of data cleaning algorithms.

In this sense, the proposed CBR contributes to conceptual framework and data cleaning ontology in order to advise the suitable data cleaning algorithm based on past experiences. CBR recommends the algorithm for each data cleaning task of the conceptual framework and it uses the ontology for case representation and recommendation of similar algorithms to the suggested. Fig. 2 depicts the CBR integration with previous works.

3. CBR for data cleaning in classification and regression tasks

The purpose of our case-based reasoning (CBR) system is to recommend data cleaning algorithms automatically to the data analyst aiming at preparing the data for classification and regression tasks. Fig. 3 presents the CBR proposed.

First, we explain the case-base construction based on meta-features of the dataset, followed by the retrieval phase where the most similar case to a new case is retrieved. Subsequently, considering a data cleaning ontology, in the reuse phase, we suggest similar solutions to the solution space found. In the retain phase, we consider three data quality dimensions for case retention (Accuracy, Completeness, and Validity).

3.1. Case-base construction

We defined a case as an ordered pair $(\rho, \mu(\rho))$ in which ρ is a problem, and $\mu(\rho)$ is the solution associated to the problem. In our approach, we represent the problem space by dataset meta-features, attributes, and target variables.

The problem space is described by dataset meta-features. Several works focused on meta-learning [32–39] have been defined dataset and attribute meta-features. Based on these works, we used twelve meta-features to describe the dataset and nine meta-features to represent each attribute of the dataset (numeric or nominal, respectively). Table 1 presents meta-features found in the meta-learning works.

The meta-features missing values and correlation correspond to numeric and nominal attributes. Three meta-features are used for numeric attributes: candidate outliers, kurtosis and skewness. Concerning nominal attributes, three features are defined: normalized entropy, mutual information and labels. Same attribute meta-features are used for dependent variable (numerical or nominal). Additionally, for nominal dependent variables, we use the imbalance ratio to measure the classes distribution.

Table 1
Meta-features used for representing problem space.

Problem space type	Meta-feature	Reference
Dataset	Instances	[32–36]
Dataset	Attributes	[32–36]
Dataset	Data dimensionality	[32–36]
Dataset	Missing values ratio	[35]
Dataset	Duplicate instances ratio	[39]
Dataset	Mean absolute linear correlation	[32–34]
Dataset	Equivalent number of features	[32–34]
Dataset	Mean absolute skewness	[32–34]
Dataset	Mean absolute kurtosis	[32–34]
Dataset	Mean attribute entropy	[32–34]
Dataset	Mean mutual information	[32–34]
Dataset	Noise-signal ratio	[32–34]
Attribute	Missing values	[35]
Attribute	Correlation	[32–34]
Attribute	Candidate outliers	[32–36]
Attribute	Skewness	[32–34,37]
Attribute	Kurtosis	[32–34,37]
Attribute	Normalized entropy	[32–34,37]
Attribute	Mutual information	[32–34,37]
Attribute	Labels	[35,38]
Attribute	Imbalance ratio	[38]

The solution space contains the algorithms used to clean each dataset. We represent the cases through data cleaning ontology explained in Section 2.2. Fig. 4 presents an example of case representation through ontology.

In Fig. 4, we present an example of a case of Polish Companies Bankruptcy dataset [40]. The dataset instances: DS1_PolishCompaniesBankruptcy and attribute: DS1_att1, DS1_att5 represent the case space problem, while Data Cleaning Algorithm instances indicate the case solution (Local Outlier Factor, Smote, Sequential Backward Elimination, ListWise Deletion and Bayesian linear regression).

In order to create case-base, we collected the datasets from UCI Repository of Machine Learning Databases [41] from the last twenty years (1998–2018). We selected datasets for classification and regression tasks, 36 cases for classification and 20 for regression tasks, totaling 56 cases.

3.2. Case retrieval

We propose a case retrieval mechanism composed of a filter and similarity phases. In the first phase, we defined two filter approaches based on clustering and quartile. These filters retrieve a reduced number of relevant cases. The second phase, computes a ranking of recovered cases by filter approaches and generates similarity scores between the new case and the retrieved cases. In the second phase, we proposed two similarity mechanisms based on meta-features of dataset and attributes. Fig. 5 presents the case retrieval architecture.

3.2.1. Filter phase

This phase retrieves the relevant cases concerning the new case. We proposed two filter methods:

Case clustering

The purpose of case clustering is to group cases into subsets called clusters. Therefore, the similar cases are grouped in the same cluster. Thus, given a new case C_q, this one is classified in a Cluster $Cluster_i$ when it has a high degree of similarity in respect to the case stored into Cluster $Cluster_i$. We used k-means as a cluster algorithm, a popular partition method widely used in the data mining community [42,43].

Please cite this article as: D.C. Corrales, A. Ledezma and J.C. Corrales, A case-based reasoning system for recommendation of data cleaning algorithms in classification and regression tasks, Applied Soft Computing Journal (2020) 106180, https://doi.org/10.1016/j.asoc.2020.106180.

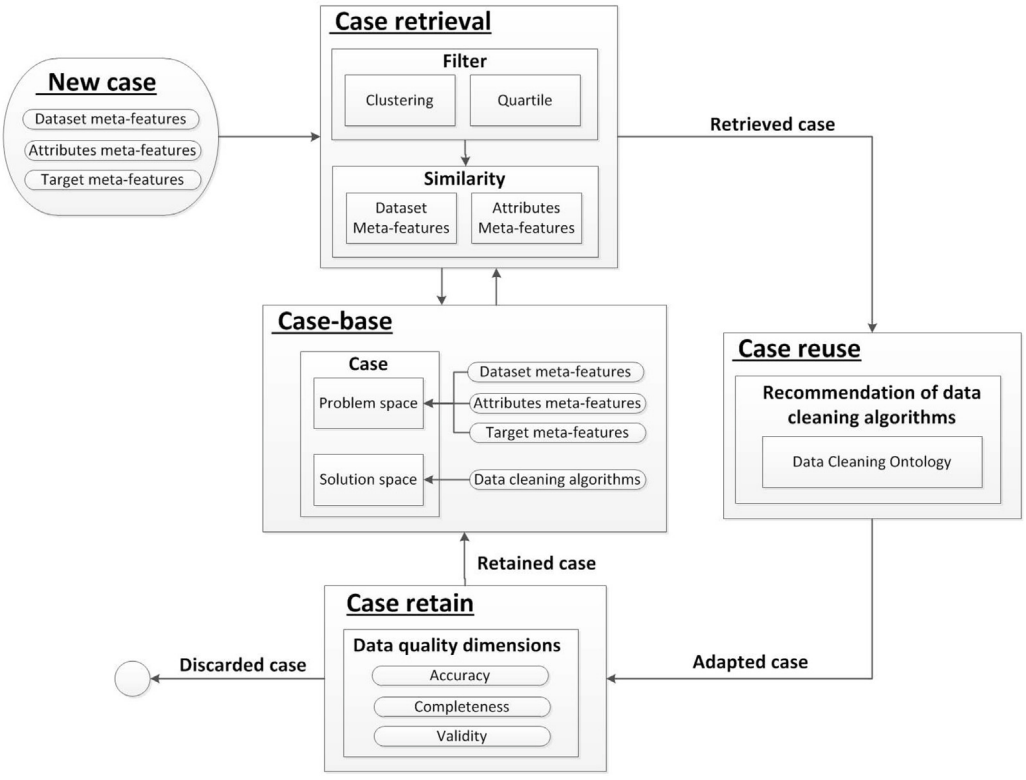


Fig. 3. CBR for data cleaning in knowledge discovery tasks.

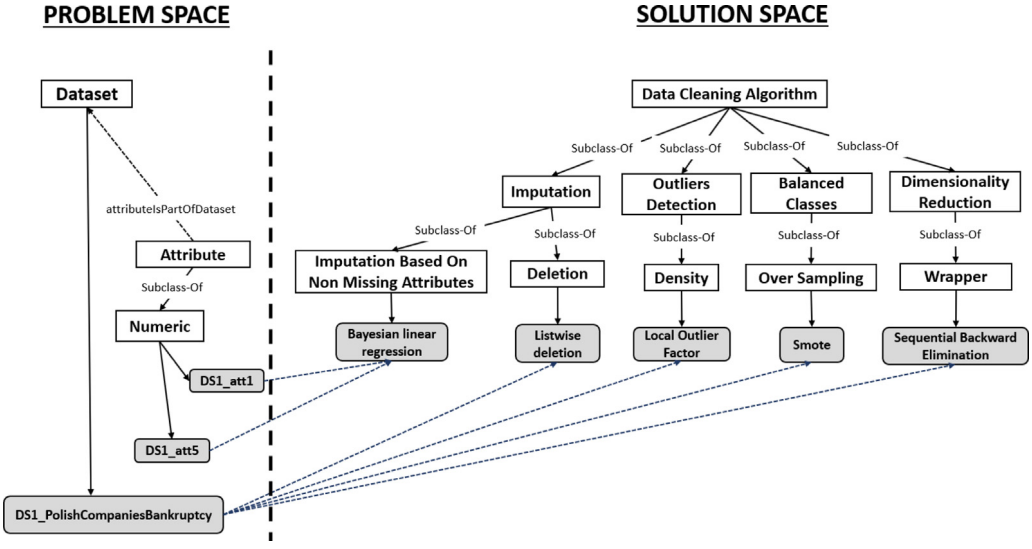


Fig. 4. Example of case representation through data cleaning ontology for the dataset of Polish companies bankruptcy. Gray square represents the class individuals while white square depicts the classes. The solid line means a hierarchical relation and the dotted line indicates the data cleaning algorithms used in the dataset and attributes.

We tested the space problems of the cases using k-means with 2, 3, 4, 5, 6 and 7 clusters, for classification and regression cases. Figs. 6a and 6b present the cases distribution in the clusters.

To classify a new case C_q in a specific cluster, we built a decision tree C4.5 and Multilayer Perceptron (MLP) from Weka tool for 2, 3, 4, 5, 6 and 7 clusters. We used the default experimental configuration of Weka to build these classifiers. As validation method, we used cross validation with 10 folds.

In this case, we are interested in assessing the proportion of cases that belong correctly to a cluster. Thus, we used the True Positive (TP) Rate as performance measure. Figs. 7a and 7b present the True Positive (TP) Rate for the obtained models.

We selected the models with the highest true positive (TP) rate for classification and regression tasks (Figs. 7a and 7b). Subsequently, we analyzed the variability of the observations within each cluster through measure within-cluster sum of squares (WCSS). Small sum of squares represents compact clusters while clusters with large sum of squares exhibit greater variability of the observations.

MLP with 6 and 2 clusters were the models with the highest TP rate for classification tasks (99.8%). We selected MLP with 6

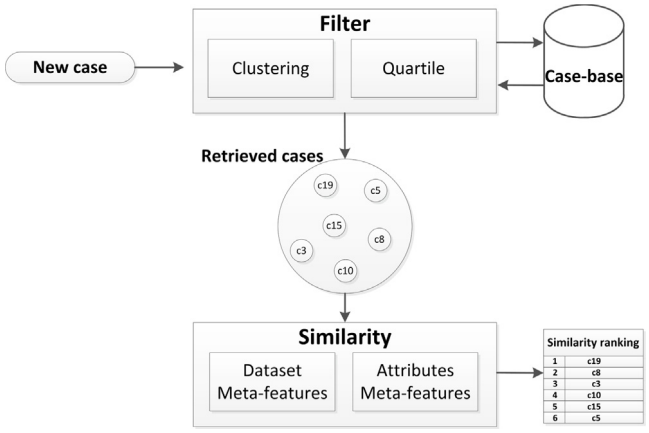


Fig. 5. CBR for data cleaning in knowledge discovery tasks.

clusters due WCSS (11) is less than WCSS of MLP with 2 clusters (30.16). Concerning to regression tasks, C4.5 with 4 clusters and MLP with 2 clusters achieved the highest TP rate (95%). We selected C4.5 with 4 clusters due this model reached lowest WCSS (6.71) compared with MLP with 2 clusters (15.28).

Case quartile

Quartiles extract fundamental information about a variable distribution that complements other traditional metrics like the mean, mode, and standard deviation [44]. We apply the quartile analysis to the dataset features defined in Section 3.1. Fig. 8 shows an example of quartile analysis for 12 cases arranged by Missing values ratio, Mean absolute kurtosis and Mean attribute entropy.

Thus, a new case C_q is classified in a quartile according to values of the dataset features. In the example of Fig. 8, C_q is classified in Q_2 of Missing values ratio, Q_1 of Mean absolute kurtosis and Q_3 of Mean attribute entropy. Finally, the cases C_{10}, C_{12}, C_2, C_5, C_6, C_8, C_4 of the quartiles Q_2, Q_1 and Q_3 (omitting the duplicate cases) are the most similar cases in respect to C_q.

Aiming at selecting the best filter mechanism to reduce the search space, in Section 4, we present the analysis of the two filter approaches evaluation for classification and regression tasks, respectively.

3.2.2. Similarity mechanisms

This phase computes a similarity ranking of the retrieved cases by filter approaches. We proposed two similarity mechanisms, the first one based on dataset meta-features, and the second one on meta-features of dataset attributes.

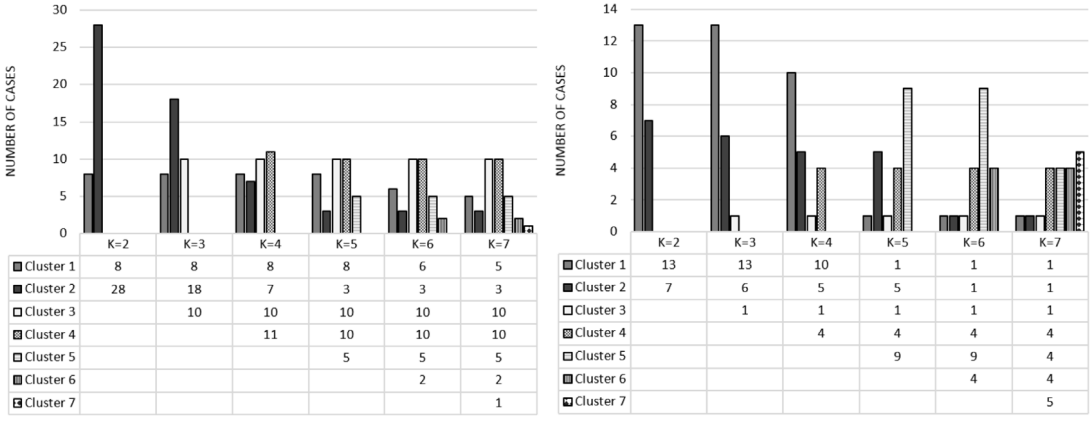
Similarity based on dataset meta-features — Sim(ds), the attribute-value representation of a case is defined as vector of dataset meta-features (Section 3.1): $C_i = [metFeat_1, metFeat_2, \ldots, metFeat_n]$ where i represents the *ith* case. Therefore, the assessment of similarity between two cases C_q and C_t is given by:

1. The similarity between values of attributes (local similarities) illustrated in Eq. (1).

$$Sim_{metFeat_j}(C_q(metFeat_j), C_t(metFeat_j)) \quad (1)$$

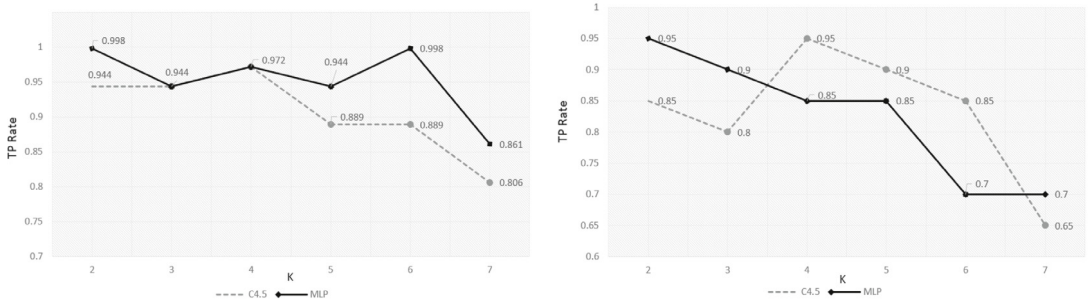
where C_q is the query case, C_t the target case, and j the *jth* feature.

2. The global similarity between C_q and C_t cases shown in Eq. (2). This measure consists of a sum of local similarity



(a) Classification tasks (b) Regression tasks

Fig. 6. Cases distribution in the clusters.



(a) Classification tasks (b) Regression tasks

Fig. 7. True positive rate of C4.5 and MLP for 2, 3, 4, 5, 6 and 7 clusters.

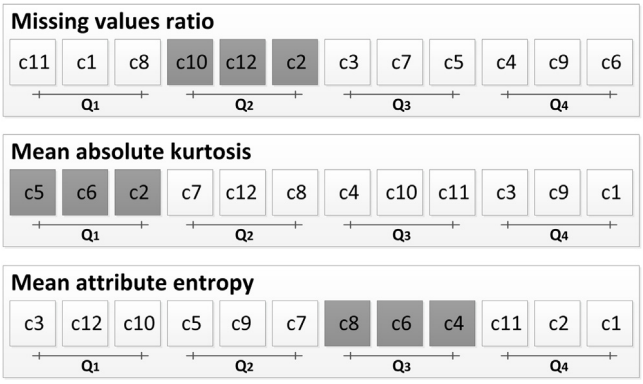


Fig. 8. Quartile analysis for missing values ratio, Mean absolute kurtosis and mean attribute entropy. The gray cells correspond to quartiles where the new case C_q is classified.

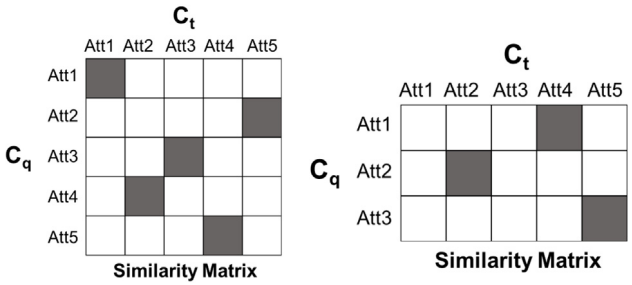


Fig. 9. Attribute matching for Exact (a) and Plugin (b) categories. Rows represent the dataset attributes of C_q, while the columns depict the dataset attributes of C_t. Gray cells represent the highest similarity for each attribute of C_q respect to C_t attributes.

measures and assumes a limited value between 0 and 1.

$$\sum_{j=1}^{n} W_j * Sim_{metFeat_j}(C_q(metFeat_j), C_t(metFeat_j)) \quad (2)$$

where W_j is the weight of the *jth* feature.

The choice of similarity measures is mostly ad hoc and there is no principle behind it. Based on study of measures for attribute-value representations [45], we used the similarity measures: arithmetic, euclidean, canberra. Subsequently, we tested these measures and we selected the best similarity measure for each meta-feature.

Concerning the dataset normalized features (missing values ratio, duplicate instances ratio, mean absolute linear correlation and mean attribute entropy), we use the weighted Euclidean measure [46] depicted in Eq. (3) as local similarity function:

$$1 - \sqrt{\sum_{j=1}^{n} W_j * (C_q(metFeat_j) - C_t(metFeat_j))^2} \quad (3)$$

For the non-normalized features of the dataset with high dispersion, as instances, attributes, data dimensionality and mean absolute skewness, the equivalent number of features and noise-signal ratio, we use the weighted Canberra similarity [47], due to the fact that this measure is sensitive to proportional differences and it allows to identify deviations from normal observations. The weighted Canberra is defined in Eq. (4).

$$1 - \sum_{j=1}^{n} W_j * \frac{|C_q(metFeat_j) - C_t(metFeat_j)|}{|C_q(metFeat_j)| + |C_t(metFeat_j)|} \quad (4)$$

For the remaining non-normalized features (mean absolute kurtosis and mean mutual information), where the standard deviations are low, we used the arithmetic summation-based similarity presented in Eq. (5).

$$1 - \sum_{j=1}^{n} W_j * \frac{|C_q(metFeat_j) - C_t(metFeat_j)|}{Max(C_t(metFeat_j)) - Min(C_t(metFeat_j))} \quad (5)$$

where $Max(C_t(metFeat_j)) \neq Min(C_t(metFeat_j))$

The second approach named *Similarity based on meta-features of dataset attributes — Sim(att)*, we built an attribute-value method based on attributes meta-features and the target variable of a dataset (Section 3.1). The attribute-value approach is represented by a vector of dataset attributes and target variable $C_i = [numAtt_1, \ldots, numAtt_n, nomAtt_1, \ldots, nomAtt_n, target]$. In this one, the numeric attribute *numAtt* represents the set of features: *outliers*, *kurtosis*, and *skewness*; while the attribute *nomAtt* represents the features: *entropy*, *mutual information*, and *labels*. Additionally, the numeric or nominal attributes share the features: *missing values* and *correlation*. Also, the three features: *outliers*, *kurtosis*, and *skewness*, together, represent a *numeric target variable*; while *nominal target variable* is represented by two features: *entropy* and *labels*.

This attribute-value approach was implemented using a Global Similarity Function (GSF), it integrates the similarity measures of numeric and nominal attributes, and the target variable given by Eq. (6).

$$\beta_1 simNumAtt(C_q, C_t) + \beta_2 simNomAtt(C_q, C_t) + \rho simTarget(C_q, C_t) \quad (6)$$

where β_1, β_2, and ρ represents the weights of each similarity function. Below, we explain how the similarity measures of attributes and the target variable are calculated:

Similarity between attributes

First, we compared the number of numeric and nominal attributes of C_q and C_t through attribute matching:

- Exact: the number of attributes (between numeric or nominal) of C_q is equal to the number of attributes of C_t (Fig. 9a).
- Plugin: the number of attributes (between numeric or nominal) of C_q is less than the number of attributes of C_t (Fig. 9b).
- Subsume: the number of attributes (between numeric or nominal) of C_q is greater than the number of attributes of C_t (Fig. 10).

Once the attribute matching was defined, we computed the similarity for each attribute (between numeric or nominal) of C_q against all attributes of C_t, then the results were stored in a similarity matrix. Subsequently, we selected the highest similarity obtained by each attribute of C_q in respect to C_t attributes, where each attribute of C_t must be different for each attribute of C_q. Fig. 9 presents an example of attribute matching for *Exact* and *Plugin* categories, where the gray cells represent the highest similarity for each attribute of C_q in respect to C_t attributes.

Referring to *Subsume* attribute matching (Fig. 10), the number of attributes of C_q is greater than the number of attributes of C_t; a C_t attribute can be used several times to calculate the similarity between C_q attributes. Therefore, we calculated the transpose of the similarity matrix, after that, we selected the highest similarity obtained by each attribute of C_t in respect to C_q attributes. Also, we defined a penalization $\alpha = da/C_q(atts)$, where da is the number of discarded attributes of C_q for computing similarities and $C_q(atts)$ the attributes of C_q. Fig. 10 presents an example

Please cite this article as: D.C. Corrales, A. Ledezma and J.C. Corrales, A case-based reasoning system for recommendation of data cleaning algorithms in classification and regression tasks, Applied Soft Computing Journal (2020) 106180, https://doi.org/10.1016/j.asoc.2020.106180.

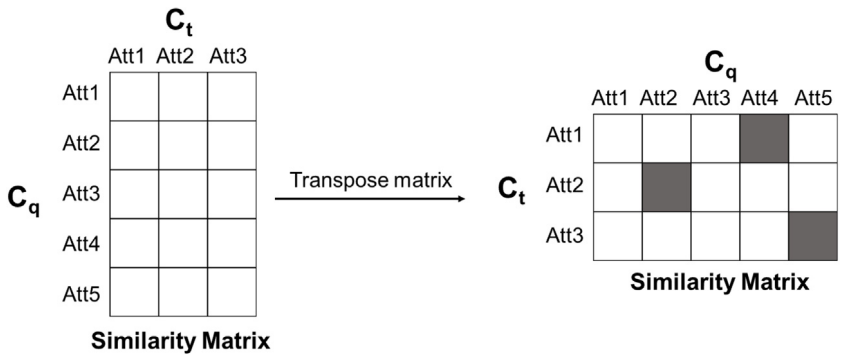


Fig. 10. Attribute matching: Subsume. Rows of the first matrix represent the dataset attributes of C_q, while the columns depict the dataset attributes of C_t. Second matrix is the transpose of similarity matrix. Gray cells represent the highest similarity for each attribute of C_q respect to C_t attributes.

Table 2
Similarity functions used in features of attributes and target variable.

Variable	Feature	Similarity function
Attribute	Correlation	Arithmetic
	Missing values	Euclidean
Numeric attribute	Candidate outliers	Euclidean
	Kurtosis	Canberra
	Skewness	Canberra
Nominal attribute	Normalized entropy	Euclidean
	Mutual information	Arithmetic
	Labels	Canberra
Target variable	Missing values	Euclidean
Numeric target variable	Candidate outliers	Euclidean
	Kurtosis	Canberra
	Skewness	Canberra
Nominal target variable	Normalized entropy	Euclidean
	Labels	Canberra

of *Subsume* attribute matching, where C_q and C_t have 3 and 5 attributes, respectively, with a penalization $\alpha = 0.4$.

Finally, the highest similarities of numeric and nominal attributes are averaged.

Similarity between target variables

We calculate the similarity of the numeric (*outliers*, *kurtosis*, *skewness*) and nominal (*entropy* and *labels*) features of C_q and C_t through local functions. We used as local similarity functions the Euclidean, Canberra and Arithmetic distance. The process to calculate the similarity between target variables follows three steps: (i) if the feature is normalized, we use Euclidean distance, (ii) if the feature is non-normalized and it has high dispersion, the Canberra distance is used and (iii) otherwise, we use Arithmetic distance. Table 2 presents the similarity functions used in the features of attributes and target variable.

In Section 4, we present the results of the filter approaches and similarity mechanisms.

3.3. Case reuse

We propose a procedure for the *Case reuse*. Given that C_t (data cleaning algorithms) is a solution for C_q, the system adjusts the C_t case as a solution for C_q. If the problem space of case C_q is equal to the problem space of C_t (which is supposed to have been successful), then the old data cleaning solution is copied [45]. If the problem space of C_q is different to C_t, the data cleaning solution is adapted and recorded before reusing it, in order to ensure the best solution for the new data quality issues. We proposed a *Data Cleaning Ontology* [31] as a recommendation mechanism of similar data cleaning algorithms to the algorithms proposed in the case solution of C_t. Fig. 11 depicts the taxonomy of dimensionality reduction algorithms of the *Data Cleaning Ontology*.

Fig. 11 depicts only the taxonomy of dimensionality reduction algorithms of the Data Cleaning Ontology. In [31], we described the complete data cleaning ontology. The ontology individuals (blue circles in Fig. 11): Information Gain, Gain Ratio, Pearson Correlation, Symmetrical Uncertainty or Chi-Squared correspond to dimensionality reduction algorithms based on filter approach. For instance, assuming that the solution of the adapted case C_t was Information Gain (filter algorithm), the Data Cleaning Ontology presents to the user, similar filter algorithms as Gain Ratio, Pearson Correlation, Symmetrical Uncertainty or Chi-Squared.

3.4. Case retain

The *Case retain* step stores the C_q case (dataset meta-features and data cleaning solution) into the temporary case-base for future reuse. The solution of the adapted case must be tested before saving it in the case-base. We reviewed approaches for the evaluation of adapted cases [45]:

1. *Human experts* who review the validity of data cleaning methods applied. The disadvantages of this approach are, time availability and vulnerability to make mistakes. These problems can be improved if experts are replaced by a formal process based on documentation (research books and papers, technical reports, etc.).
2. Evaluation of the adapted case solution in the *real world*. Results of the application of data cleaning algorithms in classification and regression tasks can provide us with feedback from reality.

Although the evaluation of adapted cases by human experts is a complex process since the verification of each new case takes a long time (we must prevent bad solutions from being retained), we consider human experts as the best evaluation approach in the real world because the second approach evaluates the adapted case after being applied in the real world. Therefore, we propose to verify the quality of the C_q case through human experts supported in three data quality dimensions:

1. *Completeness* verifies that the case has all required parts (data quality issues and data cleaning solutions) [48].
2. *Validity* is the degree to which the case conforms to a set of rules, represented within a defined data domain (e.g., if a dataset does not contain missing values, then the imputation algorithms are not used) [49].
3. *Accuracy* refers to when the data cleaning algorithms of the case solution were applied to dataset and the model generated by the cleaned dataset obtains good results [50]. The

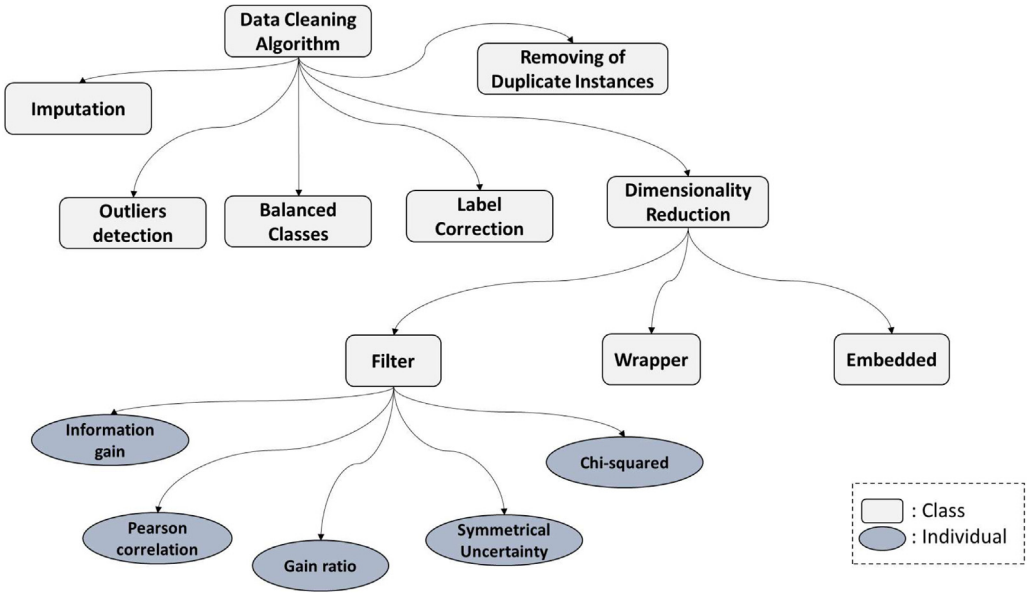


Fig. 11. Representation of dimensionality reduction algorithms in data cleaning ontology. Blue circles represent class individuals while gray squares depict classes. Solid line indicates a hierarchical relation.

Table 3
Judges experience in data mining projects.

Judge Id	Data mining experience	Years of experience
1	Master and PhD thesis	5 years
2	Teacher	3 years
3	PhD thesis	3 years
4	PhD thesis	2.5 years
5	Master thesis	2 years
6	Master thesis	2 years

measurement of Accuracy depends highly on experts. They must verify the models performance based on statistical measures, their knowledge, and the domain [45].

4. Experimental results

CBR is essentially centered on cases retrieval mechanism. The case retrieval is considered a key phase in CBR since it establishes the foundation for the general performance of CBR systems [51]. The aim of retrieval mechanisms is to retrieve the most similar case that can be successfully used to solve a new problem. If the retrieval mechanism fails, the CBR system will not produce good solutions for the new problem. Thus, we focus on the evaluation of the case retrieval mechanism proposed in Section 3.2. We used a collaborative evaluation methodology [52] which is composed of two steps: judges evaluation, and review of judges evaluation.

In the first step, a panel of judges assesses the retrieval mechanism. Table 3 presents the panel of judges and their experience in data mining projects.

The panel of judges scores the similarity between a query case against all cases of the case-base. The judges compared the meta-features values (dataset and depend variable) of each query case versus the cases contained in the case-base. Subsequently, they defined a similarity score given by value between 0%–100%. This process is addressed through evaluation forms designed in Microsoft Excel as shown Fig. 12.

For each query case, an evaluation form is designed. We defined three kinds of queries for each knowledge discovery task (classification and regression):

1. Query 1: corresponds to a copy of a case contained in the case-base. This query verifies the minimum quality of the retrieval mechanism. The retrieval mechanism and panel of judges should obtain 100% of similarity for Query 1 in respect to an identical case contained in the case-base.
2. Query 2: a modified case of a case contained in the case-base. The retrieval mechanism and panel of judges should obtain a high similarity between Query 2 and the non-modified case of the case-base.
3. Query 3: a new case that is not contained in the case-base. The aim of this query is to simulate the retrieval mechanism behavior in the real world.

The results considered relevant by the panel of judges will be the ones that represent the ideal responses for each query case.

In the second step, we compared every judges evaluation stated in the previous stage (through standard deviation — SD of the similarity scores delivered by panel of judges). If a judge evaluations are more spread out than other judges evaluations, we discarded the dispersed evaluations. In our experiment, judge 4 evaluations were discarded due they differ 40% of similarity scores SD of judges panel. Subsequently, the selected evaluations were averaged and we generated a ranking of cases.

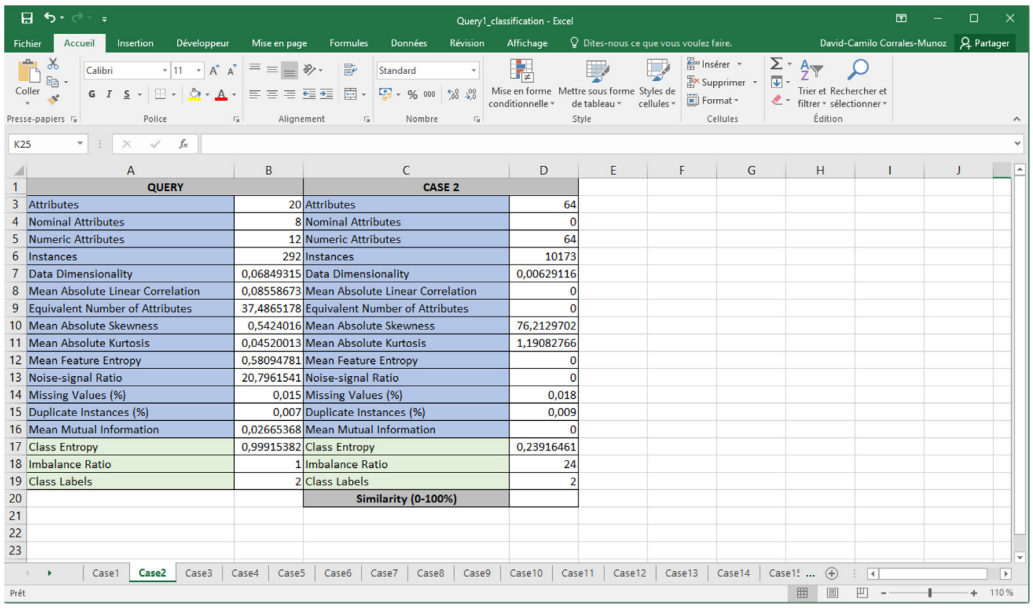
Finally, the ranking of cases proposed by the panel of judges is compared to the ranking of cases obtained by our case retrieval mechanism. To evaluate the quality of the ranking generated by our retrieval mechanism, we used two measures of retrieval information [53,54]:

- Precision@K: proportion of retrieved cases relevant in the judges ranking of K positions. Precision@K is presented in Eq. (7).

$$P@K = \frac{Rel_{cases}}{K} \tag{7}$$

  where Rel_{cases} is the number of relevant cases and K the ranking size.
- P-Precision@K: proportion of relevant retrieved cases in the same positions of the judges ranking Top-K. This measure is defined in Eq. (8).

$$P - Precision@K = \frac{P - Rel_{cases}}{K} \tag{8}$$



	A	B	C	D
1	QUERY		CASE 2	
3	Attributes	20	Attributes	64
4	Nominal Attributes	8	Nominal Attributes	0
5	Numeric Attributes	12	Numeric Attributes	64
6	Instances	292	Instances	10173
7	Data Dimensionality	0,06849315	Data Dimensionality	0,00629116
8	Mean Absolute Linear Correlation	0,08558673	Mean Absolute Linear Correlation	0
9	Equivalent Number of Attributes	37,4865178	Equivalent Number of Attributes	0
10	Mean Absolute Skewness	0,5424016	Mean Absolute Skewness	76,2129702
11	Mean Absolute Kurtosis	0,04520013	Mean Absolute Kurtosis	1,19082766
12	Mean Feature Entropy	0,58094781	Mean Feature Entropy	0
13	Noise-signal Ratio	20,7961541	Noise-signal Ratio	0
14	Missing Values (%)	0,015	Missing Values (%)	0,018
15	Duplicate Instances (%)	0,007	Duplicate Instances (%)	0,009
16	Mean Mutual Information	0,02665368	Mean Mutual Information	0
17	Class Entropy	0,99915382	Class Entropy	0,23916461
18	Imbalance Ratio	1	Imbalance Ratio	24
19	Class Labels	2	Class Labels	2
20			Similarity (0-100%)	

Fig. 12. Evaluation form (Excel file) to score the similarity between a query case against all cases of the case-base. Columns A, B show the meta-features of query case and columns C, D represent the meta-features of case contained in case-base. Blue cells indicate dataset meta-features and green cells depict the meta-features of dependent variable. The similarity score is filled in cell D20. Excel sheets contain the meta-features of each case contained in case-base and query case.

where $P - Rel_{cases}$ is the number of relevant cases located in the same positions of the judges ranking and K the ranking size.

4.1. Classification

For the case-base of classification tasks, we used the following query cases:

- *Query 1 — Autism spectrum disorder in children* is a copy of a case contained in the case-base and it describes children screening data for autism spectrum disorder [55].
- *Query 2 — Portuguese bank telemarketing (modified)* is a modified case of the case-base. We deleted three attributes and 39,000 instances. This query is related to direct marketing campaigns (phone calls) of a Portuguese banking institution [56].
- *Query 3 — Income prediction* corresponds to a new case. This query represents the income of a person in the United States that exceeds 50,000 USD per year based on census data [57].

For each query case, we applied filter approaches (Clustering and Quartile) to obtain the most similar cases. Fig. 13a presents the number of retrieved cases by filter approach. Clustering filter retrieves 5 cases for all queries, while the quartile approach retreives 33 cases for Query 1 (Q1), 30 for Query 2 (Q2) and 29 for Query 3 (Q3). In other words, clustering is a rigorous filter because it retrieves 13.88% of the cases while quartile retrieves more than 80% of the cases which can be irrelevant cases.

To verify the precision of the retrieved cases by filter approaches, in Fig. 13b we present the Precision@K with P@3, P@7, and P@10. Respecting P@3, filter approaches retrieve 100% of relevant cases for all queries. The quartile filter reaches the highest precisions for P@7 in Q1 (100%) and Q2 (85.7%), and the clustering filter by Q3 (85.7%). The quartile filter obtains the highest precision in P@10 for Q1 (90%) and Q2 (90%) and Q3 (80%). The highest precisions were obtained by the quartile filter because this approach retrieves a larger number of cases compared to the clustering filter.

Table 4
Top5 — P-Precision@K (%) for filter approaches and similarity mechanisms in classification tasks.

Query	Approach	P-P@1	P-P@2	P-P@3	P-P@4	P-P@5
Q1	Quartile - Sim(ds)	100	100	100	100	80
	Quartile - Sim(att)	100	100	100	100	80
	Cluster- Sim(ds)	100	100	100	100	80
	Cluster- Sim(att)	100	100	100	100	80
Q2	Quartile - Sim(ds)	100	100	66.70	50	40
	Quartile - Sim(att)	100	100	66.70	50	40
	Cluster- Sim(ds)	100	100	66.70	50	40
	Cluster- Sim(att)	100	100	66.70	50	40
Q3	Quartile - Sim(ds)	100	50	33.33	25	20
	Quartile - Sim(att)	100	50	66.70	50	60
	Cluster- Sim(ds)	100	50	33.33	25	40
	Cluster- Sim(att)	100	50	66.70	50	40

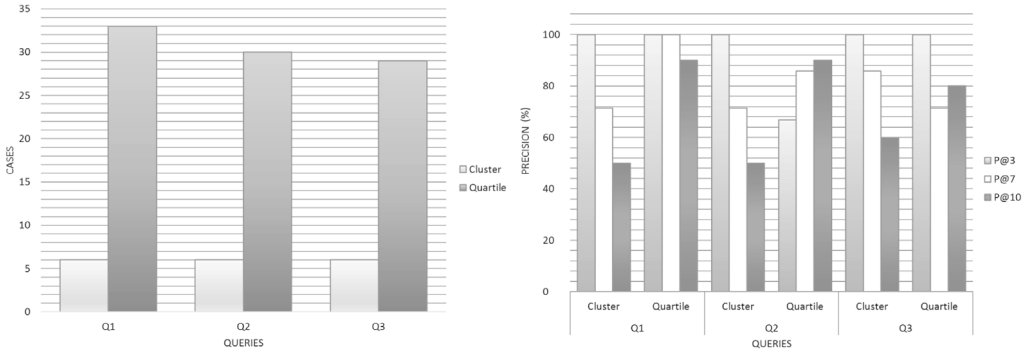
To evaluate the ranking quality of the filter approaches and similarity mechanisms, in Table 4, we show P-P@1, P-P@2, P-P@3, P-P@4, and P-P@5 for Q1, Q2, and Q3.

Filters and similarity mechanisms reach 100% of precision in P-P@1 for all queries, P-P@2 for Q1, Q2, P-P@3 and P-P@4 for Q1. These results mean that our approaches retrieve correctly the first two positions of the judges ranking for queries Q1, Q2, Q3, and the top three and four positions for Q1. In regard to P-P@5, the highest precisions are achieved in Q1 by all approaches (80%), and Q3 by the quartile approach using a Sim(att) mechanism (60%).

In general, we consider the clustering filter suitable for classification tasks, since this filter retrieves 5 cases from which 3 cases are relevant in top-3, in contrast to the quartile approach, which retrieves a large number of irrelevant cases. In respect to similarity mechanisms, they achieve the same precision for Q1 and Q2. However, in Q3 Sim(att) obtains highest precisions in P-P@3, P-P@4 and P-P@5, which means that Sim(att) is closer to the judge ranking than Sim(ds).

4.2. Regression

For the case-base of regression tasks, we used the following query cases:



(a) Retrieved cases (b) Precision P@3, P@7, and P@10

Fig. 13. Retrieved cases and P@Precision for filter approaches in classification tasks.

Table 5
Top5 — P-Precision@K (%) for filter approaches and similarity mechanisms in regression tasks.

Query	Approach	P-P@1	P-P@2	P-P@3	P-P@4	P-P@5
Q1	Quartile - Sim(ds)	100	100	100	100	80
	Quartile - Sim(att)	100	100	100	100	80
	Cluster- Sim(ds)	100	100	100	100	80
	Cluster- Sim(att)	100	100	100	100	80
Q2	Quartile - Sim(ds)	100	100	66.70	50	40
	Quartile - Sim(att)	100	100	66.70	75	60
	Cluster- Sim(ds)	100	100	66.70	50	40
	Cluster- Sim(att)	100	100	66.70	75	60
Q3	Quartile - Sim(ds)	100	50	66.70	50	40
	Quartile - Sim(att)	100	100	66.70	50	60
	Cluster- Sim(ds)	100	50	66.70	50	40
	Cluster- Sim(att)	100	100	66.70	50	40

- *Query 1 – Air pollution – benzene estimation* is a case-base case. It contains information of a gas multi-sensor device deployed on the field in an Italian city [58].
- *Query 2 — Rental bikes hourly* is a modified case of the case-base. We deleted one attribute and 8500 instances. This query contains the hourly count of rental bikes during years 2011 and 2012 at Capital bikeshare system [59].
- *Query 3 — Coffee rust* is a new case that is not included in the case-base. This query addresses coffee rust detection in Colombian crops [60].

Fig. 14a presents the number of retrieved cases by filter approaches in the case-base of regression tasks. Similar to classification tasks filters, the clustering approach retrieves a suitable number of cases compared to the quartile approach. The clustering filter retrieves 10 cases for Q1, Q2, and 4 cases for Q3, while the quartile filter retrieves 19 cases for Q1, 16 for Q2 and all cases (20) of the case-base for Q3.

With this in mind, we present in Fig. 13b the precision (P@3, P@7, and P@10) of cases retrieved by filter approaches. For Q1, the filter approaches retrieve 100% of relevant cases in P@3, P@7, and P@10. In Q2, the quartile filter achieves the highest precision for P@3 (100%), while in P@7 (85.70%) and P@10 (90%) the filter approaches reach the same precision. For Q3, the quartile filter retrieves 100% of relevant cases in P@3, P@7, and P@10 since this filter retrieves all cases of the case-base.

Finally, to evaluate the ranking quality of the filter approaches and similarity mechanisms in regression tasks, in Table 5, we present P-Precision@K for top five positions.

The cases retrieved by the filter approaches and similarity mechanisms of Q1 show 100% of precision in P-P@1, P-P@2, P-P@3, P-P@4, and 80% of precision for P-P@5. Likewise, in Q2 all filter approaches and similarity mechanisms for P-P@1, P-P@2 achieve 100% of precision, while P-P@3 achieves 66.70% of precision for all approaches. The highest precision in P-P@4 (75%), and P-P@5 (60%) are reached by filter approaches using Sim(att). Regarding Q3, P-P@1 reaches 100% of relevant cases for all approaches, while the filter methods using Sim(att) reach 100% of precision in P-P@2. The highest precision in P-P@3 (66.70%) and P-P@4 (50%) is achieved by all approaches, for P-P@5, the quartile filter and Sim(att) reach the highest precision with 60%.

In summary, the clustering filter retrieves a suitable number of cases for adaptation phase in CBR. Therefore, CBR final users have a reduced number of similar cases compared to the quartile filter. The clustering filter retrieves in average 6/36 cases for classification tasks and 10/20 cases for regression tasks, while the quartile filter considers most of cases, for example, in classification tasks it retrieves 30/36, while in regression tasks 19/20 cases.

For similarity mechanisms, precisions are equal. However, Sim(att) achieves best-ranking quality where the queries are new cases (Query 3 for classification and regression tasks).

5. Conclusions and future works

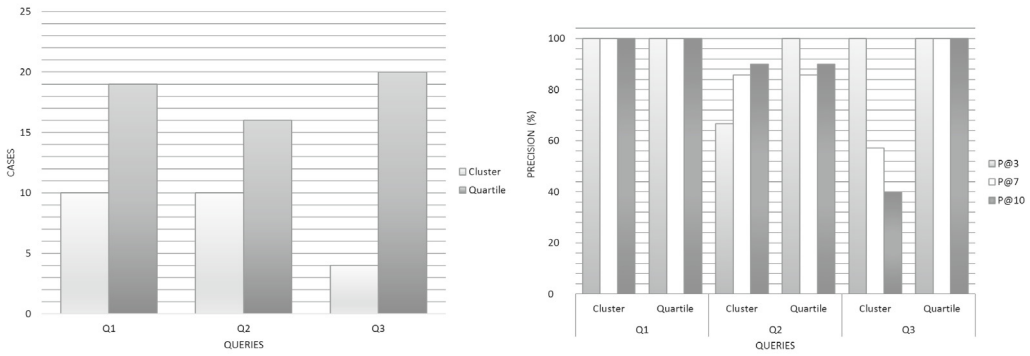
Most commercial knowledge discovery tools do not offer any system for the recommendation of data cleaning algorithms. This fact draws the authors attention [26–29] where they mentioned a list of relevant decisions that must be considered through a knowledge discovery process:

- How to effectively perform data quality verification?
- How to efficiently perform the data preparation phase (i.e. missing values, outliers, duplicate records)?
- Which data cleaning algorithm is most appropriate?
- How to deal with a potential class imbalance problem?
- How to improve the accuracy rate (i.e. error rate)?

To address the mentioned problems, we proposed a CBR system for the recommendation of data cleaning algorithms in classification and regression tasks. We considered dataset meta-features and data quality issues (missing values, outliers, imbalance classes, duplicate and contradictory instances, and high dimensionality).

Our CBR was designed based on the phases of the traditional CBR cycle (retrieval, reuse, revise, and retain), of which we focused on case retrieval mechanisms, since they require careful attention given the fact that retrieved cases contain data cleaning algorithms to recommend to users [16,19,61]. The retrieval mechanisms results (filter approaches and similarity methods) reach an average precision on judges ranking of 94.5% in top 3 (P@3), for top 7 (P@7) 84.55%, while in top 10 (P@10) 78.35%.

Please cite this article as: D.C. Corrales, A. Ledezma and J.C. Corrales, A case-based reasoning system for recommendation of data cleaning algorithms in classification and regression tasks, Applied Soft Computing Journal (2020) 106180, https://doi.org/10.1016/j.asoc.2020.106180.



(a) Retrieved cases (b) Precision P@3, P@7, and P@10

Fig. 14. Retrieved cases and P@Precision for filter approaches in regression tasks.

Concerning precision in respect to positions of judges ranking (P-Precision@K), our retrieval mechanisms for classification and regression tasks achieve 100% of precision for the first position of judges ranking and 75% for top 2 of judges ranking. The 25% of lost precision for Top 2 corresponds to irrelevant cases retrieved for the Queries 3 (new cases). These results are due to the fact that Queries 3 lack similar cases into the case-base (36 cases for classification case-base and 20 for regression).

Case-base has a low number of cases due to limited data availability, and dataset selection restrictions (each of the selected datasets must publish results of the classification or regression models used). These limitations occur at similar areas of study, for example, the CBR case-base for selection of classification and regression models contains 80 cases [22]. Other fields of study encounter similar limitations like [62], which presented a CBR for the diagnosis of gastrointestinal cancer with a case-base containing 53 cases. The CRB proposed in [63] for construction costs of multi-family housing complexes, has a case-base composed of 99 cases. While the CBR for web service discovery and selection developed in [64] counts on a case-base of 62 cases.

We propose as future works:

- Increase the number of cases. This work is intricate; however, as a first approximation, we suggest to include datasets with unpublished results (in this paper we only used dataset published in conferences and journals). In consequence, the solution spaces of new cases must guarantee high performance in the evaluation metrics (accuracy, precision, recall, mean absolute errors, etc.).
- Add other popular knowledge discovery tasks such as clustering and association rules to the CBR. This implies, to create new case-bases and define new meta-features for the new knowledge discovery tasks.
- In the retain phase, before saving the case into the case-base, we propose to build a formal process for quality assessment of cases through methodologies as [65]. The main advantage of using this methodology is the flexibility for identifying poor quality cases through a set of phases.
- Use a clustering approach based on incremental learning to avoid the update of cluster models at the filter phase [66].
- Include planners in the retrieval phase in order to build partial solutions based on a set of dataset meta-features and the knowledge represented by the Data Cleaning Ontology [11].

Declaration of competing interest

No author associated with this paper has disclosed any potential or pertinent conflicts which may be perceived to have impending conflict with this work. For full disclosure statements refer to https://doi.org/10.1016/j.asoc.2020.106180.

CRediT authorship contribution statement

David Camilo Corrales: Conceptualization, Methodology, Software, Data curation, Validation, Writing - original draft, Visualization, Investigation. **Agapito Ledezma:** Supervision, Writing - review & editing. **Juan Carlos Corrales:** Supervision, Writing - review & editing.

Acknowledgments

The authors are grateful to the research groups: Control Learning Systems Optimization Group (CAOS) of the Carlos III University of Madrid and Telematics Engineering Group (GIT) of the University of Cauca for the technical support. In addition, the authors are grateful to COLCIENCIAS for PhD scholarship granted to PhD. David Camilo Corrales. This work has been also supported by:

- Project "Alternativas Innovadoras de Agricultura Inteligente para sistemas productivos agrícolas del departamento del Cauca soportado en entornos de IoT" financed by Convocatoria 04C-2018 "Banco de Proyectos Conjuntos UEES-Sostenibilidad" of Project "Red de formación de talento humano para la innovación social y productiva en el Departamento del Cauca InnovAcción Cauca", ID-3848.
- The Spanish Ministry of Economy, Industry and Competitiveness (Projects TRA2015-63708-R and TRA2016-78886-C3-1-R).

References

[1] David Camilo Corrales, Juan Carlos Corrales, Agapito Ledezma, How to address the data quality issues in regression models: A guided process for data cleaning, Symmetry 10 (4) (2018).
[2] David Camilo Corrales, Agapito Ledezma, Juan Carlos Corrales, A conceptual framework for data quality in knowledge discovery tasks (FDQ-KDT): A proposal, J. Comput. 10 (6) (2015) 396–405.
[3] David Camilo Corrales, Agapito Ledezma, Juan Carlos Corrales, A systematic review of data quality issues in knowledge discovery tasks, Rev. Ing. Univ. Medel. 15 (28) (2016).
[4] E Soundararajan, JVM Joseph, C Jayakumar, M Somasekharan, Knowledge discovery tools and techniques, Recent Adv. Inf. Technol. (2005) 141.
[5] Usama Fayyad, Gregory Piatetsky-Shapiro, Padhraic Smyth, Knowledge discovery and data mining: Towards a unifying framework, in: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, in: KDD'96, AAAI Press, 1996, pp. 82–88.
[6] Michael R. Berthold, Nicolas Cebron, Fabian Dill, Thomas R. Gabriel, Tobias Kötter, Thorsten Meinl, Peter Ohl, Christoph Sieb, Kilian Thiel, Bernd Wiswedel, KNIME: The konstanz information miner, in: Christine Preisach, Hans Burkhardt, Lars Schmidt-Thieme, Reinhold Decker (Eds.), Data Analysis, Machine Learning and Applications, Springer Berlin Heidelberg, Berlin, Heidelberg, 2008, pp. 319–326.
[7] Markus Hofmann, Ralf Klinkenberg, RapidMiner: Data Mining Use Cases and Business Analytics Applications, Chapman & Hall/CRC, 2013.

[8] George Fernandez, Statistical Data Mining Using SAS Applications, Second Edition, second ed., CRC Press, Inc., USA, 2010.
[9] R. Baruti, Learning Alteryx: A Beginner's Guide to Using Alteryx for Self-Service Analytics and Business Intelligence, Packt Publishing, Limited, 2017.
[10] D. Cook, Practical Machine Learning with H2O: Powerful, Scalable Techniques for Deep Learning and AI, O'Reilly Media, Incorporated, 2016.
[11] Floarea Serban, Joaquin Vanschoren, Jörg-Uwe Kietz, Abraham Bernstein, A survey of intelligent assistants for data analysis, ACM Comput. Surv. 45 (3) (2013) 31:1–31:35.
[12] Besim Bilalli, Alberto Abelló Gamazo, Tomàs Aluja Banet, Robert Wrembel, Towards intelligent data analysis: the metadata challenge, in: Proceedings of the International Conference on Internet of Things and Big Data, 2016, pp. 331–338.
[13] Cullen Schaffer, A conservation law for generalization performance, in: Proceedings of the Eleventh International Conference on International Conference on Machine Learning, in: ICML'94, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1994, pp. 259–265.
[14] David H. Wolpert, The supervised learning no-free-lunch theorems, in: Rajkumar Roy, Mario Köppen, Seppo Ovaska, Takeshi Furuhashi, Frank Hoffmann (Eds.), Soft Computing and Industry: Recent Applications, Springer London, London, 2002, pp. 25–42.
[15] I. Nonaka, I. Nonaka, I.N.H. Takeuchi, P.K.I. Nonaka, H. Takeuchi, T. Hirotaka, Takeuchi, The Knowledge-Creating Company: How Japanese Companies Create the Dynamics of Innovation, in: Everyman's library, Oxford University Press, 1995.
[16] Agnar Aamodt, Enric Plaza, Case-based reasoning: Foundational issues, methodological variations, and system approaches, AI Commun. 7 (1) (1994) 39–59.
[17] David B. Leake, Case-Based Reasoning: Experiences, Lessons and Future Directions, first ed., MIT Press, Cambridge, MA, USA, 1996.
[18] Hassan Y.A. Abutair, Abdelfettah Belghith, Using case-based reasoning for phishing detection, Procedia Comput. Sci. 109 (2017) 281–288, 8th International Conference on Ambient Systems, Networks and Technologies, ANT-2017 and the 7th International Conference on Sustainable Energy Information Technology, SEIT 2017, 16–19 May 2017, Madeira, Portugal.
[19] Aijun Yan, Kuanhong Zhang, Yuanhang Yu, Pu Wang, An attribute difference revision method in case-based reasoning and its application, Eng. Appl. Artif. Intell. 65 (2017) 212–219.
[20] Robert Engels, Planning tasks for knowledge discovery in databases; performing task-oriented user-guidance, in: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, in: KDD'96, AAAI Press, 1996, pp. 170–175.
[21] Rüdiger Wirth, Colin Shearer, Udo Grimmer, Thomas Reinartz, Jörg Schlösser, Christoph Breitner, Robert Engels, Guido Lindner, Towards process-oriented tool support for knowledge discovery in databases, in: Jan Komorowski, Jan Zytkow (Eds.), Principles of Data Mining and Knowledge Discovery: First European Symposium, PKDD '97 Trondheim, Norway, June 24–27, 1997 Proceedings, Springer Berlin Heidelberg, Berlin, Heidelberg, 1997, pp. 243–253.
[22] Guido Lindner, Rudi Studer, AST: Support for algorithm selection with a CBR approach, in: Jan M. Żytkow, Jan Rauch (Eds.), Principles of Data Mining and Knowledge Discovery: Third European Conference, PKDD'99, Prague, Czech Republic, September 15-18, 1999. Proceedings, Springer Berlin Heidelberg, Berlin, Heidelberg, 1999, pp. 418–423.
[23] M. Choinski, J.A. Chudziak, Ontological learning assistant for knowledge discovery and data mining, in: 2009 International Multiconference on Computer Science and Information Technology, 2009, pp. 147–155.
[24] Karina Gibert, Miquel Sànchez-Marrè, Víctor Codina, Choosing the right data mining technique: classification of methods and intelligent recommendation, in: International Congress on Environmental Modelling and Software, 2010.
[25] Katharina Morik, Martin Scholz, The miningmart approach to knowledge discovery in databases, in: Intelligent Technologies for Information Analysis, Springer Berlin Heidelberg, Berlin, Heidelberg, 2004, pp. 47–65.
[26] Michel Charest, Sylvain Delisle, Ofelia Cervantes, Design considerations for a CBR-based intelligent data mining assistant, in: Proceedings of the 9th Maghrebian Conference on Information Technologies (MCSEAI 2006), 2006, pp. 120–125.
[27] M. Charest, S. Delisle, O. Cervantes, Yanfen Shen, Invited paper: Intelligent data mining assistance via CBR and ontologies, in: 17th International Workshop on Database and Expert Systems Applications (DEXA'06), 2006, pp. 593–597.
[28] Michel Charest, Sylvain Delisle, Ontology-guided intelligent data mining assistance: Combining declarative and procedural knowledge., in: Artificial Intelligence and Soft Computing, 2006, pp. 9–14.
[29] Michel Charest, Sylvain Delisle, Ofelia Cervantes, Yanfen Shen, Bridging the gap between data mining and decision support: A case-based reasoning and ontology approach, Intell. Data Anal. 12 (2) (2008) 211–236.
[30] Shaker El-Sappagh, Mohammed Elmogy, AM Riad, Hosam Zaghlol, Farid A Badria, EHR data preparation for case based reasoning construction, in: International Conference on Advanced Machine Learning Technologies and Applications, Springer, 2014, pp. 483–497.
[31] David Camilo Corrales, Agapito Ledezma, Juan Carlos Corrales, From theory to practice: A data quality framework for classification tasks, Symmetry 10 (7) (2018).
[32] A. Filchenkov, A. Pendryak, Datasets meta-feature description for recommending feature selection algorithm, in: 2015 Artificial Intelligence and Natural Language and Information Extraction, Social Media and Web Search FRUCT Conference (AINL-ISMW FRUCT), 2015, pp. 11–18.
[33] Guangtao Wang, Qinbao Song, Heli Sun, Xueying Zhang, Baowen Xu, Yuming Zhou, A feature subset selection algorithm automatic recommendation method, J. Artif. Intell. Res. 47 (1) (2013) 1–34.
[34] Ciro Castiello, Giovanna Castellano, Anna Maria Fanelli, Meta-data: Characterization of input features for meta-learning, in: Vicenç Torra, Yasuo Narukawa, Sadaaki Miyamoto (Eds.), Modeling Decisions for Artificial Intelligence, Springer Berlin Heidelberg, Berlin, Heidelberg, 2005, pp. 457–468.
[35] Guido Lindner, Daimlerchrysler Ag, Rudi Studer, AST: Support for algorithm selection with a CBR approach, in: Recent Advances in Meta-Learning and Future Work, 1999, pp. 418–423.
[36] Robert Engels, Christiane Theusinger, Using a data metric for preprocessing advice for data mining applications, in: Proceedings of the European Conference on Artificial Intelligence (ECAI-98, John Wiley and Sons, 1998, pp. 430–434.
[37] Matthias Reif, Faisal Shafait, Andreas Dengel, Meta2-features: Providing meta-learners more information, in: 35th German Conference on Artificial Intelligence, Citeseer, 2012.
[38] Nele Verbiest, Enislay Ramentol, Chris Cornelis, Francisco Herrera, Improving SMOTE with fuzzy rough prototype selection to detect noise in imbalanced classification data, in: Juan Pavón (Ed.), Advances in Artificial Intelligence – IBERAMIA 2012: 13th Ibero-American Conference on AI, Cartagena de Indias, Colombia, November 13-16, 2012. Proceedings, Springer Berlin Heidelberg, Berlin, Heidelberg, 2012, pp. 169–178.
[39] L. Huang, H. Jin, P. Yuan, F. Chu, Duplicate records cleansing with length filtering and dynamic weighting, in: 2008 Fourth International Conference on Semantics, Knowledge and Grid, 2008, pp. 95–102.
[40] Maciej Zieba, Sebastian K. Tomczak, Jakub M. Tomczak, Ensemble boosted trees with synthetic features generation in application to bankruptcy prediction, Expert Syst. Appl. 58 (2016) 93–101.
[41] A. Asuncion, D.J. Newman, UCI machine learning repository. Irvine, CA: University of California, school of information and computer science, 2007, URL [http://www.ics.uci.edu/mlearn/MLRepository.html].
[42] Junjie Wu, Advances in K-means Clustering: A Data Mining Thinking, Springer Publishing Company, Incorporated, 2012.
[43] D.C. Corrales, J.C. Corrales, A. Sanchis, A. Ledezma, Sequential classifiers for network intrusion detection based on data selection process, in: 2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC), 2016, pp. 001827–001832.
[44] William Christian Krumbein, The use of quartile measures in describing and comparing sediments, Am. J. Sci. (188) (1936) 98–111.
[45] Michael M. Richter, Rosina O. Weber, Case-Based Reasoning: A Textbook, Springer Publishing Company, Incorporated, 2013.
[46] Jan De Leeuw, Sandra Pruzansky, A new computational method to fit the weighted euclidean distance model, Psychometrika 43 (4) (1978) 479–490.
[47] G.N. Lance, W.T. Williams, Mixed-data classificatory programs i - Agglomerative systems, Aust. Comput. J. 1 (1) (1967) 15–20.
[48] Matthew Bovee, Rajendra P. Srivastava, Brenda Mak, A conceptual framework and belief-function approach to assessing overall information quality, Int. J. Intell. Syst. 18 (1) (2003) 51–74.
[49] Laura Sebastian-Coleman, Measuring Data Quality for Ongoing Improvement: A Data Quality Assessment Framework, first ed., Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2013.
[50] Daniele Barone, Fabio Stella, Carlo Batini, Dependency discovery in data quality, in: Barbara Pernici (Ed.), Advanced Information Systems Engineering, Springer Berlin Heidelberg, Berlin, Heidelberg, 2010, pp. 53–67.
[51] Ramon Lopez De Mantaras, David McSherry, Derek Bridge, David Leake, Barry Smyth, Susan Craw, Boi Faltings, Mary Lou Maher, MICHAEL T COX, Kenneth Forbus, et al., Retrieval, reuse, revision and retention in case-based reasoning, Knowl. Eng. Rev. 20 (3) (2005) 215–240.
[52] Armando Ordoñez, Hugo Ordoñez, Juan Carlos Corrales, Carlos Cobos, Leandro Krug Wives, Lucinéia Heloisa Thom, Grouping of business processes models based on an incremental clustering algorithm using fuzzy similarity and multimodal search, Expert Syst. Appl. 67 (2017) 163–177.
[53] David Dolan Lewis, Representation and Learning in Information Retrieval (Ph.D. thesis), University of Massachusetts, Amherst, MA, USA, 1992, UMI Order No. GAX92-19460.

Please cite this article as: D.C. Corrales, A. Ledezma and J.C. Corrales, A case-based reasoning system for recommendation of data cleaning algorithms in classification and regression tasks, Applied Soft Computing Journal (2020) 106180, https://doi.org/10.1016/j.asoc.2020.106180.

[54] David Camilo Corrales, Jose Eduardo Gomez, Juan Carlos Corrales, Comparación Estructural y Linguistica de Procesos de Negocio Semánticos, first ed., Research and Innovation Book, 2012.
[55] Fadi Thabtah, Autism spectrum disorder screening: Machine learning adaptation and DSM-5 fulfillment, in: Proceedings of the 1st International Conference on Medical and Health Informatics 2017, in: ICMHI '17, ACM, New York, NY, USA, 2017, pp. 1–6.
[56] Sérgio Moro, Paulo Cortez, Paulo Rita, A data-driven approach to predict the success of bank telemarketing, Decis. Support Syst. 62 (2014) 22–31.
[57] Ron Kohavi, Scaling up the accuracy of Naive-Bayes classifiers: A decision-tree hybrid, in: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, in: KDD'96, AAAI Press, 1996, pp. 202–207.
[58] Saverio De Vito, Marco Piga, Luca Martinotto, Girolamo Di Francia, CO, NO2 and NOx urban pollution monitoring with on-field calibrated electronic nose by automatic bayesian regularization, Sensors Actuators B 143 (1) (2009) 182–191.
[59] Hadi Fanaee-T, Joao Gama, Event labeling combining ensemble detectors and background knowledge, Prog. Artif. Intell. 2 (2) (2014) 113–127.
[60] David Camilo Corrales, Agapito Ledezma, Andrés J Peña, Javier Hoyos, Apolinar Figueroa, Juan Carlos Corrales, A new dataset for coffee rust detection in Colombian crops base on classifiers, Sist. Telemát. 12 (29) (2014) 9–23.
[61] Sheng-Tun Li, Hei-Fong Ho, Predicting financial activity with evolutionary fuzzy case-based reasoning, Expert Syst. Appl. 36 (1) (2009) 411–422.
[62] Renata Saraiva, Mirko Perkusich, Lenardo Silva, Hyggo Almeida, Clauirton Siebra, Angelo Perkusich, Early diagnosis of gastrointestinal cancer by using case-based and rule-based reasoning, Expert Syst. Appl. 61 (2016) 192–202.
[63] Joseph Ahn, Moonseo Park, Hyun-Soo Lee, Sung Jin Ahn, Sae-Hyun Ji, Kwonsik Song, Bo-Sik Son, Covariance effect analysis of similarity measurement methods for early construction cost estimation using case-based reasoning, Autom. Constr. 81 (2017) 254–266.
[64] Alan De Renzis, Martin Garriga, Andres Flores, Alejandra Cechich, Alejandro Zunino, Case-based reasoning for web service discovery and selection, Electron. Notes Theor. Comput. Sci. 321 (2016) 89–112, CLEI 2015, the XLI Latin American Computing Conference.
[65] Yosef Jabareen, Building a conceptual framework: philosophy, definitions, and procedure, Int. J. Qual. Methods 8 (4) (2009) 49–62.
[66] S. Young, I. Arel, T.P. Karnowski, D. Rose, A fast and stable incremental clustering algorithm, in: 2010 Seventh International Conference on Information Technology: New Generations, 2010, pp. 204–209.

Please cite this article as: D.C. Corrales, A. Ledezma and J.C. Corrales, A case-based reasoning system for recommendation of data cleaning algorithms in classification and regression tasks, Applied Soft Computing Journal (2020) 106180, https://doi.org/10.1016/j.asoc.2020.106180.