A New Metric for Measuring the Intrinsic Quality in Data Collected for Quantitative Classification

Roxane Jouseau[1], Sébastien Salva[1], and Chafik Samir[1]

LIMOS - UMR CNRS 6158, Clermont Auvergne University, France
sebastien.salva@uca.fr, chafik.samir@uca.fr,
roxane.jouseau@doctorant.uca.fr

Abstract. Learning an optimal classification model intrinsically depends on data quality. Despite many efforts for its characterization, existing methods have often limited quality measures to specific criteria, leading to the lack of comprehensive definitions and rigorous formulations. Indeed, its evaluation is related to the context and often requires external elements, which implies a process that is long and prone to errors. Therefore, there is still a strong need for solutions that enable effective data quality assessment.
This paper addresses the resulting scientific challenges and introduces a new metric, specifically designed for numerical classification problems. Unlike existing measures, the proposed solution is based on the correlated evolution between classification performance and data deterioration. Therefore, it offers three main advantages: Being model independent, not requiring the use of external reference data while offering a solution that is easy to adapt for several real-world scenarios. Additionally, we provide a comprehensive interpretation of the quality scores and illustrate the main evaluation levels with use cases. We demonstrate its effectiveness through extensive experiments and comparisons with the state of the art.

Keywords: Classification · Numeric Dataset · Data Quality · Machine Learning · Metric.

1 Introduction

Data plays a central role in many industrial and scientific fields [36]. However, generating, collecting, and storing data does not guarantee an optimal use. Indeed, several recent studies have highlighted the substantial loss that companies can incur due to poor data quality. This has a significant impact on cost, decision making and even customer satisfaction [31]. As a result, assessing the quality of data before allocating resources to analyze and use it becomes crucial.
To achieve an accurate assessment of quality, it is essential to introduce rigorous definitions and concepts that take into account relevant measures and appropriate tools. In previous works, data quality has often been limited to the impact of certain standard criteria, usually called quality *dimensions*. These serve as

guidelines for evaluating assumptions or criteria, either objective or subjective, in a specific context. For example, several works have focused on aspects such as accuracy, completeness, reliability, usefulness or timeliness for particular applications and contexts [16]. These criteria have often been evaluated separately or with a combination of scores that is difficult to justify. Hence, a first limitation concerns the absence of a rigorous definition and, consequently, a precise measurement.
In addition, the assessment of data quality is often linked to a specific context which quite often requires external metadata, rules and reliable references. Establishing or extracting these external elements is generally a time-consuming, expertise-dependent and error-sensitive process [12]. The need for external metadata to assess data quality reveals a second limitation and highlights the complexity of providing in-depth assessments in their absence. This has been confirmed by recent studies which revealed the scarcity of tools capable of assessing data quality effectively [12].
In this work, we deal with these limitation by introducing a new data quality metric, adapted to learning models for numerical classification problems. From the user point of view, the proposed metric takes data as input and returns a score as well as a quality level (good, average or bad). This without requiring external data such as metadata, rules to model the prior knowledge, or reliable references. The proposed metric offers, among others, a notable advantage by providing a consistent way to assess the quality of various types of datasets: Varying number of classes, domains and dimensionalities. We have also paid a special attention to define a metric that is easy to use, model independent and easily interpretable.
To best clarify the context, we first report on previous works that have attempted to propose definitions and measures for data quality. Subsequently, we present the foundations of the proposed methods and describe the main contributions.

1.1 Related Work

Data Quality Dimensions Several papers introduce data quality measures and their importance for specific applications and contexts. Some surveys have attempted to extract a general quality definition along with the most used dimensions [12,35,8,5].

In [35], different types of data quality assessments are categorized into objective or subjective dimensions (in [12], the terms hard and soft measurements are used instead). The subjective dimensions, e.g., Timeliness or Trust, which reflect the needs of stakeholders, have domain-specific definitions that differ too much to allow comparison between different datasets. On the other hand, the objective dimensions, e.g., Accuracy or Validity, refer to quantifiable aspects that assess the quality of the data. These dimensions are highly dependent on the context, which is often defined using rules formulated by specialists in the domain of the data at hand. Even though data contexts are mostly static, the formulation of dynamic contexts was also proposed in [27]. The need for data contexts and metadata, which are often long to write and prone to errors, entails that these

dimensions are not simple to use or interpret and cannot be generalized since different definitions are given.

The main dimensions used to characterize data quality, such as Accuracy, Completeness, Consistency, and Timeliness, can be found in recent publications [12,8,5]. Yet, there is no unified definitions of these concepts. For example, Accuracy definitions in the social and applied sciences rely on multiple notions [17]. Although there is agreement among definitions that Accuracy can be characterized using the concept of error magnitude, this consensus does not extend to Accuracy measures. For instance, the authors of [12] identified three commonly used measure definitions. The same authors also focused on Completeness. They note that Completeness is usually defined as the "breadth, depth, and scope of information contained in the data" on the condition that the data exists [42,6,12]. The most generic measure of Completeness is the number of complete elements over the total number of elements. However, some minor variations are proposed in [20,3]. Consistency usually aims to evaluate semantic rule violations over tuples or relational tables [6]. These rules can be evaluated either with boolean as proposed in [20] or fuzzy logic or, in some situations, can even be evaluated over time by comparison to past data instances as suggested in [39]. Timeliness refers to the temporal relevance of the information contained in a dataset and also feeds on the notions of currency and volatility [6]. A discussion of various definitions and measures for Timeliness can be found in [19], yet there is no consensus on a unified definition. Within our context of classification tasks, Class Imbalance, which refers to a situation where the distribution of instances across different classes is not uniform, is another dimension often employed to evaluate data quality.

While there are numerous dimensions discussed in the literature, many of them are context-dependent and necessitate additional information before providing accurate measurements. Consequently, comparing different datasets is still challenging.

Data Quality Tools The authors of [12] conducted a survey of 667 software tools purportedly dedicated to data quality. Surprisingly, they identified only 11 tools that provide information about data quality, as opposed to focusing on data visualization and profiling: Apache Griffin [15], Ataccama ONE [4], DataCleaner [10], Datamartist [11], Experian Pandora [13], InformaticaDQ [22], InfoZoom and its extension IZDQ [23], MobyDQ [37], OpenRefine [33] and its extension MetricDoc [7], SAS Data Quality [38], and Talend Open Studio [41].

The data quality measures proposed by these tools were classified into the 4 following categories: Accuracy, Completeness, Consistency, Timeliness, and others. The authors concluded that most of the tools provide one or two measures that assist data engineers in data profiling only. They mainly focus on easing the definition of data indicators and assisting data profiling. The authors also put aside tools that rely on rules, since their writing require expert knowledge, which is often not available. As a result, they only kept four tools: Apache Griffin, InformaticaDQ, MobyDQ, and MetricDoc. Among these tools, we observed that

three of them require external data. Apache Griffin and MobyDQ take as input a reference dataset, making them less practical as ground truth reference datasets are not always available. MetricDoc, which allows the computation of a ratio of duplicates at the table level, asks for the user's criteria to express what a duplicate is. InformaticaDQ focuses on textual data, such as elements of postal addresses, email addresses, etc., and is irrelevant to numeric datasets, which is what our work focuses on. Finally, we tried to investigate the Data Quality toolset by IBM [21]. However, despite taking the steps to access the free trial version on their website, we could not secure working access to their API, which is a problem that the authors of [12] also seem to have faced, as mentioned in their paper.

In summary, few tools are available to assist data engineers in measuring data quality. Most of these tools rely on a limited set of data dimensions, often requiring external data. Currently, no publicly available tool offers a straightforward method for measuring data quality across various datasets.

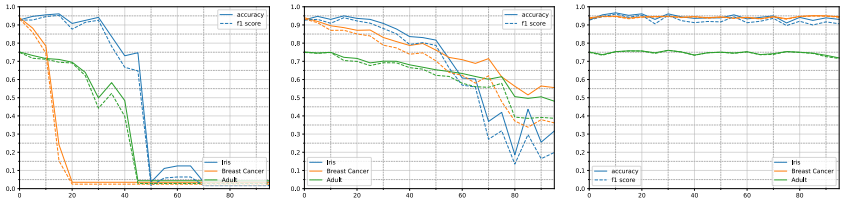
The Impact Of Data Quality On Performance Data quality evaluation also indirectly appears when model performance is measured, e.g., with accuracy or f1-score in the context of classification tasks [16]. Several studies have focused on performance for observing the impact of some specific errors and their detection and repair in this context [28,2,32,24]. The method presented in [9] estimates the limiting performance of a classifier imposed by a database and its quality defaults. Even though this method could be considered a premise of the current one, it treats data as a static element and does not offer insight into identifying the data quality issues that create these limitations.

These papers showed that data quality has a significant impact on the performance of machine learning models. Besides, machine learning models trained on high-quality data are generally more robust to generalization.

1.2 The Proposed Metric

These observations motivated us to present in [26] a preliminary work introducing a data quality measure for numeric classifications. This measure was a special case of the more general metric covered in this study. The measure is based on two concepts that allow to evaluate data quality, without the need of dimension. Given a dataset, we firstly measure performance across a wide range of classification models. Furthermore, we assess variations of performance when data is deteriorated, i.e., when a small percentage of errors is injected into the dataset in a controlled way. We indeed observed that the evolution of the performance of a dataset that is slightly deteriorated incrementally does not follow a linear curve, as we would expect. Instead, the performance evolution makes some distinctive features stand out, which we shall capture by means of our metric.

Figure 1 exemplifies these concepts on three well-known datasets Iris [14], (Breast) Cancer [43], and Adult [1]. We chose 12 different classification models available in scikit-learn [34]: Logistic regression, K-Nearest Neighbors, Decision



(a) Missing values injected in training sets (b) Outliers injected in training sets (c) Partial duplicates injected in training sets

Fig. 1: Evolution of the mean accuracies (solid lines) and f1-scores (dashed lines) when errors are injected in Iris, Cancer, and Adult (missing values (a), outliers (b), and fuzzing (c)).

tree, Random forest, Ada boost, Naive Bayes, XGboost, Support vector classification, Gaussian process, Multi-layer perceptron, Stochastic gradient descent, and Gradient boosting. We also selected three different types of errors: missing values, outliers, and fuzzing, a.k.a. partial duplicates. In Figures 1a, 1b, and 1c, we depict the evolution of the mean accuracies (solid lines) and f1-scores (dashed lines) over 30 iterations of injecting controlled percentages of errors randomly generated with a uniform distribution, in training data. We inject up to 95% of errors with a 5% increment. The 12 classification models are then trained on these deteriorated datasets with a random split using 80% of the data for training and 20% for testing. Two main observations can be made from these figures:

- **Observation 1**: accuracy and f1-score decrease with data quality when errors are injected into training data. The extent of this decrease varies among classification models and across the datasets. This observation is illustrated in Figures 1a and 1b. For instance, in Figure 1a, for the dataset Iris, the mean accuracy stays over 0.8 up to the injection of 35% of missing values into training data. We can observe that by injecting 0 to 30% of errors, the performance remains good. But after 35%, the mean accuracy curve drops down. With 40% of missing values, the dataset seems to be of medium quality. For Breast Cancer, the curve drops when 10% of errors are injected. This observation tends to suggest that the quality of Breast Cancer is lower than the quality of Iris. This is what we intend to capture with our metric in a more precise way.
- **Observation 2:** the decrease in accuracy and f1-score are nonlinear. They are low when data is of good quality or when data is extremely deteriorated. However, the decrease in performance is significantly high between these two states. This performance behavior is especially visible in Figure 1a. We can observe that the mean accuracy or f1-score drops down significantly with only a 5% increment of errors. This performance drop reflects quality issues. Our metric is designed to evaluate data quality between these two states.

On the other hand, some error types have little to no impact on model performances. For instance, Figure 1c shows that despite the incremental fuzzing of the training data, the model performances stay quite steady. This observation must also be encoded by our metric.

1.3 Contributions

We introduce a novel data quality metric DQ for numeric data within the scope of classification tasks. DQ aims at measuring the quality of a dataset, through model performance evaluation across a selection of classification models, along with performance variations with a small dataset deterioration. To assist data engineers in their quality analyses, we provide an interpretation of the DQ measurements that categorizes a dataset as either of good, medium, or bad quality. Ultimately, we offer an algorithm and tool for assessing data quality on a given dataset, regardless of whether a trusted test set is provided (a test set, which has been curated and verified to ensure that it can effectively be used to measure performance).

The first benefit of DQ is its ability to facilitate the comparison of different datasets of varied dimensions, number of classes, number of attributes, and domains of application, as it does not use dimensions that can be interpreted differently with regard to the context. Another important benefit is that our metric does not require external data (metadata, rules, datasets, etc.), making it highly user-friendly and applicable in real-life scenarios where access to such information is often challenging. This paper also provides an empirical evaluation on 210 datasets, which investigates the effectiveness of DQ to return accurate quality scores with or without trusted test set. We also compare DQ with some classical quality dimensions, namely: Completeness, Class Imbalance, and Data Dimensionality. We also investigate the efficiency of a prototype tool implementing the computation of DQ.

DQ extends the first measure based on accuracy we proposed in [26] by incorporating a list of measures each related to any performance indicator. This offers the advantage for DQ to be versatile, i.e. to be effectively applicable in a wider range of scenarios. Furthermore, this paper presents extensive experiments to investigate four criteria related to the effectiveness and efficiency of the metric under various conditions.

In summary, the major contributions of this paper are:

- the presentation of DQ, a novel data quality metric for numeric datasets in the context of classification tasks. DQ corresponds to a vector of measures that allow taking into account varied performance indicators,
- the interpretation of DQ with the proposition of thresholds and of three straightforward quality levels (good, medium, bad),
- an algorithm for computing DQ on a dataset with or without trusted test set,
- a tool publicly available in [25] to compute DQ on real datasets,
- an evaluation of DQ to investigate its effectiveness and efficiency.

Paper organization: Section 2 presents our data quality metric DQ. We define DQ independently to classification performance and show how to apply it by considering two performance measures, accuracy and f1-score. We also discuss the impact of having trusted test sets or not on the quality evaluation and present an algorithm for computing DQ. In Section 3, we propose an interpretation of DQ with regard to accuracy and f1-score. We then study the effectiveness and efficiency of DQ in Section 4 through an empirical evaluation with 210 datasets. Finally, Section 5 summarizes our contributions and draws some perspectives for future work.

2 Definition Of The Metric *DQ*

DQ corresponds to a list of measures of the form $(q^{\Phi_1}, \ldots, q^{\Phi_n})$ where $\Phi_i (1 \leq i \leq n)$ are performance indicators, e.g., accuracy. The next section presents the general definition of q^{Φ}, then we apply q^{Φ} to two classification performance indicators: accuracy and f1-score.

2.1 Definition Of q^{Φ}

We use the following notations in the remainder of the paper: D is the dataset to evaluate, the set of models is denoted M, and the set of error types is denoted E. Φ stands for a classification performance measure. q^{Φ} is composed of two formulas q_1^{Φ} and q_2^{Φ} that encode both observations discussed in Section 1.3. Specifically, given a dataset D, $q_1^{\Phi}(D)$ formalizes Observation 1, i.e., the performance evaluation across a set of classification models in M. The use of multiple classification models aims at building a metric that is not model or context-dependent. $q_2^{\Phi}(D)$ formalizes Observation 2, i.e., the evaluation of performance variations when a low percentage of errors are injected in training sets. It aims at capturing abnormally high performance variations over small dataset perturbations.

We define q^{Φ} as:

$$q^{\Phi}(D) := max(q_1^{\Phi}(D), q_2^{\Phi}(D)) \tag{1}$$

We chose to use the maximum to ensure that q^{Φ}:

- increases as data quality decreases;
- is bounded between 0 and 1;
- captures the most variation of data quality possible.

Definition Of q_1^{Φ} We denote $P(\Phi, m, D)$ the performance of the model $m \in M$ parameterized for D, with the performance indicator Φ. The average performance $P_M(\Phi, D)$ is defined as:

$$P_M(\Phi, D) := \frac{1}{|M|} \sum_{m \in M} P(\Phi, m, D) \tag{2}$$

We at least expect to get an average performance $P_M(\Phi, D)$ greater than the performance of the random classifier, which we denote α_Φ. We have:

$$0 < P_M(\Phi, D) - \alpha_\Phi \leq 1 - \alpha_\Phi \tag{3}$$

If $P_M(\Phi, D)$ is lower, we consider the quality of D to be the lowest possible. The function δ_1 captures this statement:

$$\delta_1(P_M(\Phi, D)) := \begin{cases} 1 & if\ P_M(\Phi, D) > \alpha_\Phi \\ 0 & otherwise \end{cases} \tag{4}$$

We now define $q_1^\Phi(D)$ as:

$$q_1^\Phi(D) := 1 - \frac{P_M(\Phi, D) - \alpha_\Phi}{1 - \alpha_\Phi} \delta_1(P_M(\Phi, D)) \quad (\text{with } \alpha_\Phi < 1) \tag{5}$$

We have $0 \leq q_1^\Phi(D) \leq 1$, with $q_1^\Phi(D)$ the closer to 0, the better the quality of D.

Definition Of q_2^Φ q_2^Φ captures the fact that the model performance is mostly unaffected by small data deteriorations for high or low-quality datasets, as presented in Observation 2. We evaluate this property by measuring performance variations of the classification models in M when a percentage of error is injected into the data. To avoid any bias in our metric, we assume that errors are injected randomly, with a uniform distribution in the training data. $D_{e,p}$ stands for the dataset obtained from D after injecting p percentage of error of type e.

$\Delta P_{M,e}(\Phi, D)$ expresses the variation of performance, measured with Φ for a specific error e and is defined as:

$$\Delta P_{M,e}(\Phi, D) := \frac{1}{|M|} \sum_{m \in M} abs(P(\Phi, m, D) - P(\Phi, m, D_{e,p})) \tag{6}$$

Pragmatically, we observed that a small percentage of errors p is sufficient to detect performance variations. For example, consider the dataset Breast Cancer completed with 10% of missing values in Figure 1a (horizontal axis x=10). If we inject 5% of errors in this dataset, the performance of the resulting model (horizontal axis x=15) drops significantly. This result suggests that the dataset Breast Cancer completed with 10% of missing values is of bad quality. If we consider Iris (horizontal axis x=0), the injection of 5% of errors produces a new dataset whose performance is not impacted (horizontal axis x=5). This suggests that Iris is of good quality. A bigger percentage of errors would significantly affect the content of the dataset, which would be too different from the original dataset we want to evaluate.

We want $\Delta P_{M,e}(\Phi, D)$ to detect and measure significant variations of performance, as small variations are expected due to the injection of errors into D. To exclude small variations, we define δ_2 as:

$$\delta_2(\Delta P_{M,e}(\Phi, D)) := \begin{cases} 1 & if\ \Delta P_{M,e}(\Phi, D) > p \\ 0 & otherwise \end{cases} \tag{7}$$

We can now define $q_2^{\Phi}(D)$ as:

$$q_2^{\Phi}(D) := min(\frac{10}{|E|} \sum_{e \in E} \Delta P_{M,e}(\Phi, D)\delta_2(\Delta_{M,e}(\Phi, D)), 1) \tag{8}$$

We chose to add a factor of 10 in q_2^{Φ} to express that a performance variation of 10% or more when we inject a small amount of error in data is an indication of bad data quality. As this factor does not keep the result bounded by 1, we use the minimum function to define q_2^{Φ}.

2.2 Application To Accuracy And F1-score

We exemplify the concretization of DQ with two performance indicators: accuracy and f1-score. The former is widely used and easy to interpret. Still, it is also well-known that accuracy may not be a reliable measure when the data distribution is imbalanced across classes. While the f1-score is more challenging to interpret, it does not suffer from this issue.

For accuracy, the data quality of D is measured with $q^{acc}(D) = (q_1^{acc}(D), q_2^{acc}(D))$. q_1^{acc} requires to set the parameter α_{acc}, which expresses the performance achieved with a random class choice. Here, $\alpha_{acc} = 1/c$ with the number of classes in the dataset D. $q_2^{acc}(D)$ requires to set the percentage p of errors, which are injected in the dataset D, in order to compute performance variations. Pragmatically, we noticed that a small percentage of errors $p = 5\%$ is sufficient to capture these variations. A higher value of p is possible but may lead to less precise measurements of the loss of performance.

$q^{f1}(D)$ expresses the data quality of D measured by means of the f1-score. With q_1^{f1}, as previously, we want an f1-score better than the performance of a random classifier. If we denote the precision p and recall r, the f1-score is defined as $\frac{2pr}{p+r}$. Again, we consider that the f1-score should be better than the f1-score of a random classifier with a uniform distribution of the data across the classes. In other terms, we do not try to optimize recall or precision. As such, we set $p = r = \frac{1}{c}$ and we obtain the parameter α_{f1} equal to $\frac{1}{c}$ as well. For q_2^{f1}, for consistency reasons, we keep the percentage of errors $p = 5\%$.

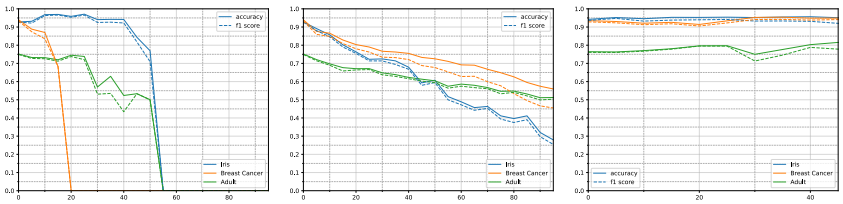
Both q^{acc} and q^{f1} are based upon the set of classification models M and the set of error types E. In the remainder of the paper, we consider that M is made up of the following 12 classification models: Logistic regression, K-Nearest Neighbors, Decision tree, Random forest, Ada boost, Naive Bayes, XGboost, Support vector classification, Gaussian process, Multi-layer perceptron, Stochastic gradient descent, and Gradient boosting. We selected these models through scikit-learn [34] to cover varied types of models. The set E contains the three error types studied previously: missing values, outliers, and fuzzing. We selected them because these are considered classical error types often encountered in numerical datasets. Besides, we observed that they have different impacts on model

performance: outliers and missing values have the most impact on accuracies and f1-scores; fuzzing tends to have less impact and offers the benefit of simulating data generation.

2.3 The Impact of not having Trusted Test Set

Our definition of the metric DQ strongly relies on the evaluation of the performance over a set of models. Most of the time, ready to use test sets are not provided with the data. Thus, a test set is usually generated by partitioning the original dataset D assuming that both the training and test sets have the same quality. To reduce the impact of such a strong hypothesis, we empirically show that our quality metric is still valid under this condition using several sub-samplings from D.

We computed the accuracy and f1-scores from Iris, Breast Cancer, and Adult after injecting controlled percentages of errors (missing values, outliers, and fuzzing) from 0% to 95% with a 5% increment. However, instead of using trusted test sets, we uniformly sampled 30 training and test subsets from D to train and test the 12 classification models listed in M. Hence, we decrease the chance of bad quality testing datasets.



(a) (b) (c)

Fig. 2: Evolution of mean accuracies and f1-scores when errors are injected in Iris, Cancer, and Adult: missing values (a), outliers (b), and fuzzing (c).

Figure 2 gives, for each error type, the mean accuracy and f1-score of the 12 classification models. If we compare these results with the performances computed on the same datasets with trusted test sets given in Figure 1, we observe that even though individual values of mean accuracy and f1-score are different, they stay in the same range. And most importantly, the overall decreases reflect the same tendencies. As a consequence, both Observations 1 and 2, along with the metric definition are still valid. We, therefore, consider that computing a mean of q^{acc} and q^{f1} over 30 re-samplings of D, when a trusted test set is not available, makes sense.

Finally, we summarize all the steps mentioned previously for evaluating the quality of D with $DQ(D)$ in Algorithm 1. The metric interpretation, mentioned in line 19 of this algorithm, is discussed in the next section.

Data: Dataset D
Result: $DQ(D)$ and its interpretation.
if *D is made up of a trusted test set* **then**
  $LD = (D)$;
else
  Generate the list $LD = (D^1, \ldots, D^{30})$ of resampled versions of D;
end
foreach $q^{\Phi} = (q_1^{\Phi}, q_2^{\Phi})$ *in* DQ **do**
  foreach $D^i \in LD$ **do**
    Compute $q_1^{\Phi}(D^i)$ as defined in Eq. 5;
    foreach *error type* $e \in$ **do**
      Create a new dataset $D^i_{e,p}$ by injecting D^i with $p\%$ of error e, randomly generated with a uniform distribution;
    end
    Compute $q_2^{\Phi}(D^i)$ with Eq.(8);
  end
  $q_1^{\Phi}(D) = AM_{(D^i \in LD)} q_1^{\Phi}(D^i)$;
  $q_2^{\Phi}(D) = AM_{(D^i \in LD)} q_2^{\Phi}(D^i)$;
  Compute $q^{\Phi}(D)$ with Eq.(1);
  Interpret $q^{\Phi}(D)$;
end

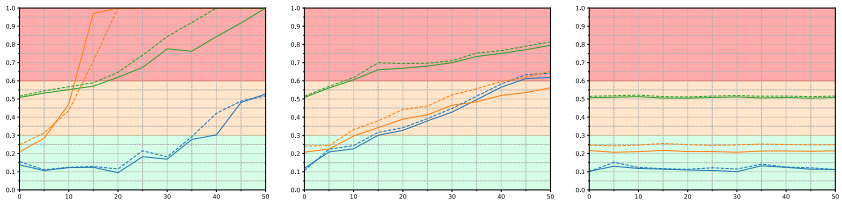
Algorithm 1: $DQ(D)$ computation

3 Interpretation of the Metric

As stated previously, DQ is a list of quality measurements $(q^{\Phi_1}, \ldots, q^{\Phi_n})$. The more diverse and numerous performance indicators $\Phi_1, \ldots, \Phi_n$ employed, the more nuanced and comprehensive the evaluation of data quality will be. As a result, it is recommended to select the most appropriate indicators for the specific targeted objectives. The interpretation of a score list returned by DQ can be achieved by studying every score and then striking a balance that aligns with these objectives. If no objective is clearly set, we suggest considering $max(q^{\Phi_1}, \ldots, q^{\Phi_n})$ to draw a data quality interpretation, capturing the worst possible data quality.

In this paper, we consider two quality measurements, q^{acc} and q^{f1}. We propose to develop a more in-depth interpretation of every measurement by empirically extracting two thresholds $th1$ and $th2$ that will allow us to express three comprehensible levels of quality: good, medium, and bad. Setting optimal thresholds is inherently difficult, as they act as boundaries that introduce uncertainty, especially for datasets whose quality measurements reside in transitional areas. Besides, predefined standards and objectives might also impact threshold choices. To stay context-free, the definition of these thresholds will be solely based on the impact of the dataset quality on the classification performance. To guide our threshold selection, we empirically derived them by studying both

measurements q^{acc} and q^{f1} on 93 datasets.



(a) (b) (c)

Fig. 3: Evolution of q^{acc} (solid lines) and q^{f1} (dashed lines) when errors are injected in training data (missing values (a), outliers (b), and fuzzing (c)).

Initial threshold observations: we first estimate the thresholds by considering the known qualities of the datasets Iris, Breast Cancer, and Adult. The first two datasets are widely accepted as having good quality in the literature, while Adult is of medium quality. We compute $q^{acc}(Iris) = 0.14$, $q^{f1}(Iris) = 0.16$, $q^{acc}(Cancer) = 0.21$, $q^{f1}(Cancer) = 0.25$, $q^{acc}(Adult) = 0.51$, and $q^{f1}(Adult) = 0.52$. It results that the threshold $th1$ should be above 0.3 and $th2$ above 0.5 for both q^{acc} and q^{f1}.

Next, we study the accuracies and f1-scores on the datasets obtained from Iris, Breast Cancer, and Adult when controlled percentages of missing values, outliers, and fuzzing are respectively injected into the training sets, in 5% increments up to 50%. Figure 1 illustrates the performance curves for these datasets. Figures 3a, 3b and 3c illustrate the evolution of q^{acc} and q^{f1} when percentages of missing values, outliers and fuzzing are injected in the original datasets.

Selection of $th2$: in Figure 1a, we observe that the quality of the datasets derived from Breast Cancer with 10 up to 15% of missing values becomes bad because of its high number of attributes combined with a relatively low number of samples (31 and 569) mean that for 10 and 15% of missing values, most samples contain at least one missing value. A close inspection of the dataset actually shows that at 10%, only 24 samples do not contain any missing values. If we observe $q^{acc}(Cancer)$ and $q^{f1}(Cancer)$ in Figure 3a, we can deduce that $th2$ should be around 0.55 or 0.6. This is confirmed with the datasets obtained from Adult after the injection of 15% of errors. Furthermore, in Figure 1b, we observe that the datasets obtained from Adult with more than 10% of outliers go to bad quality because both the accuracy and f1-score fall under 0.70. If we observe $q^{acc}(Adult)$ and $q^{f1}(Adult)$ in Figure 3b we deduce that $th2$ should be set to 0.6.

Selection of $th1$: in Figure 1a, we observe that the datasets derived from Breast Cancer with 5% and 10% of missing values go to medium quality because,

even though the accuracy and f1-scores are high, a sharp decrease in accuracy and f1-score is observed immediately with more than 10% of missing values. If we observe $q^{acc}(Cancer)$ and $q^{f1}(Cancer)$ in Figure 3a for these same percentages of errors, we deduce that $th1$ should be set to 0.3. The quality of the datasets derived from Iris when 35 and 40% of missing values are injected also turns medium because we observe significant decreases in accuracy and f1-score starting respectively from 35 and 40% of missing values. The results depicted in Figure 3a confirms that $th1$ should be set to 0.3 because $q^{acc}(Iris) = 0.3$ and $q^{f1}(Iris) = 0.29$ for 40 and 35% of missing values.

In Figure 1c, we can observe the performances of the models obtained from Iris, Breast Cancer, and Adult with the injection of percentages of fuzzing. The curves stay steady, which is expected as fuzzing does not induce a loss of information. Hence, the datasets derived from Iris and Breast Cancer must be of good quality, and those from Adult must be of medium quality. In Figure 3c, we observe that the quality scores given by q^{acc} and q^{f1} confirm these observations and do not contradict the previously established thresholds.

Proposition of quality levels: for both q^{acc} and q^{f1}, we finally propose the following interpretation thresholds:

- $q^{\Phi} \leq 0.3$: data quality is good;
- $0.3 < q^{\Phi} \leq 0.6$: data quality is medium;
- $0.6 < q^{\Phi}$: data quality is bad.

These thresholds, along with the colors representing the three quality levels, are depicted in Figure 3. We believe that these thresholds will serve as valuable guides for data engineers, aiding them in the selection of datasets that will maximize the effectiveness of classification tasks. But, such thresholds have to be seen as boundaries rather than absolute limits as they are uncertain. Pragmatically, when a quality measurement q^{ϕ} returns a value close to a threshold, data engineers should proceed to further analysis to estimate the appropriate quality level.

As mentioned previously, using several model performance measures should provide a more nuanced evaluation of data quality. Figure 4 illustrates this statement by showing both q^{acc} and q^{f1} for the datasets obtained from Iris, Cancer, and Adult after generating training sets with varying levels of class imbalance while still maintaining a balanced test set. We artificially built imbalanced datasets by incrementally removing 5% of samples in one of the classes of a dataset (always the same class). The horizontal axis expresses the obtained datasets from 0% to 95% of removed samples. Figure 4 illustrates the interest in using both accuracy and f1-score for measuring data quality. Unsurprisingly, the figure shows that q^{f1} better captures the problem of class imbalance by returning higher values than those given by q^{acc} (hence, data quality measured with q^{f1} is lower).

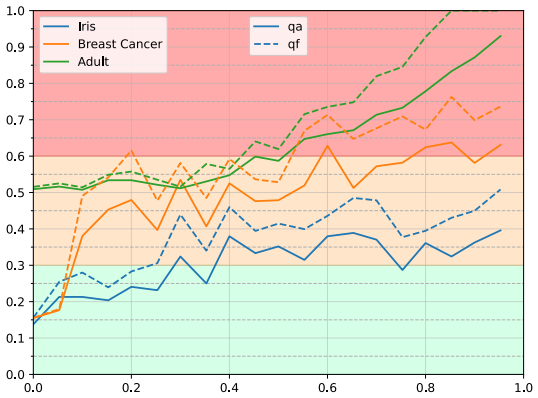


Fig. 4: Evolution of q^{acc} (solid lines) and q^{f1} (dashed lines) with imbalanced classes for Iris, Breast Cancer, and Adult. The horizontal axis shows the percentage of samples removed in one class

4 Experimental Results

To empirically evaluate the proposed metric DQ, we investigated the following four criteria (C1, C2, C3, C4):

1. The ability of DQ to correctly characterize data quality.
2. The relevance of DQ in the absence of trusted tests.
3. The comparison of DQ with quality dimensions.
4. The performance when using Algorithm 1.

These criteria were studied with the same parameters (E,M,p) discussed in Section 2.2. Before exploring these criteria, we first present the datasets used for evaluation.

4.1 Datasets Description

Real datasets. We use five different datasets, Spambase, Heart Disease, Abalone, Dry Beans, and Statlog [30], with varied number of classes, number of samples, number of attributes, and applications. This choice is motivated by the study of DQ across several real-world scenarios and conditions. Some details about these datasets are summarized in Table 1 where the last column reports data quality. We consider that Spambase is of good quality with regard to the mean accuracy 0.9 obtained with the models of M. Besides, the number of samples is relatively high (4601) for 57 attributes. The other datasets are considered of medium quality since, for each case, the mean accuracy is relatively low. Additionally, Abalone and Dry Beans present high levels of class imbalance, which are usually considered as quality issues. The mean accuracy for Dry Beans may

appear very low compared to the others but this is not the case since it is a seven-classes classification.

Table 1: Summary and details of the evaluation datasets.

Dataset	Number of classes	Samples total	Number of attributes	Features	Class imbalance Small class	Missing data	Mean accuracy	Estimated data quality
Spambase	2	4 601	57	integers, reals	1 813	None	0.90	good
Abalone	Original: 28 post-processing: 2	4 177	8	categorical, integers, reals	1 407 post-processing	None	0.84	medium
Dry Beans	7	13 611	16	categorical, integers, reals	522	None	0.68	medium
Statlog	2	Original: 1 000 post-processing: 959	23	integers	275	41	0.76	medium
Heart Disease	Original: 5 stages post-processing: 2	Original: 303 post-processing: 297	13	categorical, integers, reals	??? post-processing No	6	0.79	medium

Semi-synthetic datasets with deterioration. We created 150 variant datasets from the initial ones through artificial deterioration in a controlled way. We injected different percentages of missing values, outliers, or fuzzing separately and uniformly. The error injection was performed in 5% increments up to 50%. Additionally, we created 55 imbalanced variants by incrementally removing 5% up to 50% of samples from the same class. In summary, the experiments were performed on 210 datasets, including 205 variants. The datasets, results, and a prototype tool based on Algorithm 1 are available online [25].

4.2 C1 Ability Of DQ to Characterize Data Quality Correctly

Setup: we investigate C1 by comparing measurements and quality levels given by *DQ* with quality levels estimated manually. We considered the five datasets presented in Table 1 along with 100 deteriorated datasets. We manually estimated the quality levels of the datasets with 5%, 10%, 30%, 35%, and 50% of errors (or having a reduction of x % of samples in one of the classes of an initial dataset). These quality level estimations rely on our interpretation of 7 characteristics, which are summarized in Tables 7, 8, 9, 10, 11 and 9 given in the Appendix. The data quality level interpretations are given in the columns "E" of these tables. The datasets deteriorated with fuzzing are expected to keep the same quality level as the initial datasets since this error corresponds to the addition of partial duplicate samples.

Results: Figure 5 illustrates the *DQ* scores. We display q^{acc} as solid lines and q^{f1} as dashed lines, and we show the three quality levels in different colors. In these figures, the x-axis represents the list of datasets with x percents of deterioration. To make the comparison easier, the estimated quality levels and the computed ones with *DQ* are given in Tables 2 - 5. For instance, the first table summarizes the results for the datasets with 5, 10, 30, 35, and 50% of missing values. This table also includes a specific column "Initial" to report the results obtained from the 5 datasets before deterioration. The column "E" shows the estimated quality levels, the column q^{acc} (resp. q^{f1}) reports the quality levels

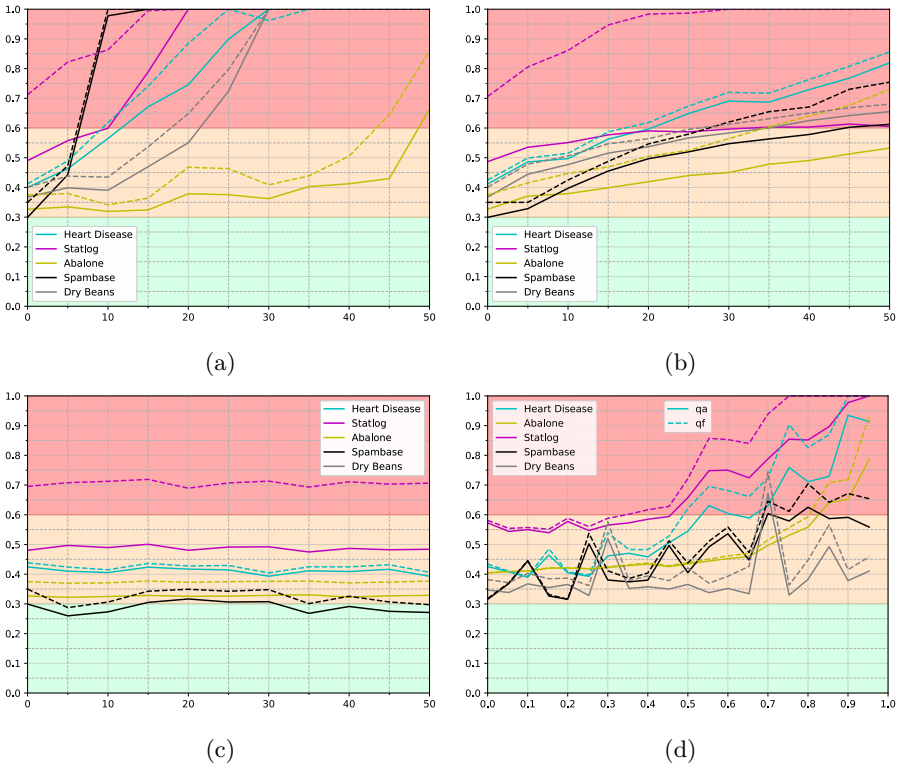


Fig. 5: q^{acc} (solid curves) and q^{f1} (dashed curves). The x-axis shows datasets with x percents of deterioration for a) missing values, b) outliers, c) fuzzing and d) class imbalance.

obtained with q^{acc} (resp. q^{f1}). The quality levels given by DQ that are different from what is expected are marked in red.

Table 2: Comparison between estimated data quality (E) and measured quality levels with q^{acc} and q^{f1} for the datasets with 0, 5, 10, 30, 35 and 50% of missing values. G,M,B stand for Good, Medium, Bad, respectively. Quality levels in red show incorrect computed levels.

Dataset	Initial			5%			10%			30%			35%			50%		
	E	q^{acc}	q^{f1}	E	q^{acc}	q^{f1}	E	q^{acc}	q^{f1}	E	q^{acc}	q^{f1}	E	q^{acc}	q^{f1}	E	q^{acc}	q^{f1}
Heart Disease	M	M	M	M	M	M	M	M	B	B	B	B	B	B	B	B	B	B
Statlog	M	M	B	B	M	B	B	M	B	B	B	B	B	B	B	B	B	B
Abalone	M	M	M	M	M	M	M	M	M	M	M	M	M	M	M	B	B	B
Spambase	G	G	M	M	M	M	B	B	B	B	B	B	B	B	B	B	B	B
Dry Beans	M	M	M	M	M	M	M	M	M	B	B	B	B	B	B	B	B	B

If both q^{acc} and q^{f1} are used together to evaluate data quality, our metric is effective at 95% to capture the correct quality levels (100 correct results for 105 datasets). If q^{acc} or q^{f1} is used separately, we capture 87% of correct quality levels with q^{acc} and 81% with q^{f1}. We now investigate the case where the quality levels returned by DQ are significantly different from our estimations (quality levels in red).

In Table 2, 2 of the 15 quality levels calculated for the initial datasets are incorrect. For Statlog, q^{f1} expresses bad quality, whereas medium quality is expected. This is because q^{f1} better detects the class imbalance problem in this dataset. We interpreted the quality level of these datasets as medium, but we could actually consider it as bad if class imbalance is central. For Spambase, q^{f1} returns medium instead of good. We can observe in Figure 5a that the values given by q^{f1} are close to the threshold. The value, hence, falls within a region of uncertainty. This problem due to thresholds chosen to separate the 3 quality levels happens several times. In Table 2, for the deteriorated datasets, we again observe that the values given by q^{acc} and q^{f1} are close to the thresholds. For example, it happens for Heart Disease with 10% of errors and for Statlog with 5% and 10%.

We studied the quality levels of the deteriorated datasets in red in Table 3. Again, several incorrect quality levels are given by DQ because the scores are close to thresholds. This translates to a zone of uncertainty for choosing between two quality levels (for q^{acc}: Statlog with 30% and 35% of outliers, Abalone with 50%; for q^{f1}: Spambase with 30% and 35% of outliers, Dry Beans with 30% and 35%). We carefully studied the datasets Statlog with 5% and 10% of outliers, and we again observed that q^{f1} better captures class imbalance.

We then studied the quality levels of the deteriorated datasets with fuzzing in Table 4. Here, we always expect to obtain the same quality levels as the original datasets for the deteriorated ones. Again, we observe that q^{f1} better detects the

Table 3: Comparison between estimated data quality (E) and measured quality levels with q^{acc} and q^{f1} for the datasets with 5,10,30,35 and 50% of outliers.

Dataset	5%			10%			30%			35%			50%		
	E	q^{acc}	q^{f1}	E	q^{acc}	q^{f1}	E	q^{acc}	q^{f1}	E	q^{acc}	q^{f1}	E	q^{acc}	q^{f1}
Heart Disease	M	M	M	M	M	M	B	B	B	B	B	B	B	B	B
Statlog	B	M	B	B	M	B	B	M	B	B	M	B	B	B	B
Abalone	M	M	M	M	M	M	M	M	M	M	M	M	B	M	B
Spambase	M	M	M	M	M	M	M	M	B	M	M	B	B	B	B
Dry Beans	M	M	M	M	M	M	M	M	B	M	M	B	B	B	B

Table 4: Comparison between estimated data quality (E) and measured quality levels with q^{acc} and q^{f1} for the datasets with 5, 10, 30, 35 and 50% of fuzzing.

Dataset	5%			10%			30%			35%			50%		
	E	q^{acc}	q^{f1}	E	q^{acc}	q^{f1}	E	q^{acc}	q^{f1}	E	q^{acc}	q^{f1}	E	q^{acc}	q^{f1}
Heart Disease	M	M	M	M	M	M	M	M	M	M	M	M	M	M	M
Statlog	M	M	B	M	M	B	M	M	B	M	M	B	M	M	B
Abalone	M	M	M	M	M	M	M	M	M	M	M	M	M	M	M
Spambase	G	G	G	G	G	M	G	G	M	G	G	M	G	G	G
Dry Beans	M	M	M	M	M	M	M	M	M	M	M	M	M	M	M

class imbalance of the datasets derived from Statlog. For Spambase with 10, 30, and 35% of fuzzing, we obtain a medium quality level instead of good, but once more the scores illustrated in Figure 5c are close to thresholds.

Table 5: Comparison between estimated data quality (E) and measured quality levels with q^{acc} and q^{f1} for datasets with one class reduced by 5, 10, 30, 35 and 50% of samples.

Dataset	5%			10%			30%			35%			50%		
	E	q^{acc}	q^{f1}	E	q^{acc}	q^{f1}	E	q^{acc}	q^{f1}	E	q^{acc}	q^{f1}	E	q^{acc}	q^{f1}
Heart Disease	M	M	M	M	M	M	M	M	M	M	M	M	B	M	B
Statlog	M	M	M	M	M	M	M	M	M	M	M	M	B	B	B
Abalone	M	M	M	M	M	M	M	M	M	M	M	M	M	M	M
Spambase	G	M	M	G	M	M	G	M	M	G	M	M	G	M	M
Dry Beans	M	M	M	M	M	M	M	M	M	M	M	M	M	M	M

We finally investigated the incorrect quality levels shown in Table 5. For Heart Disease with one class reduced by 50%, both the q^{acc} and q^{f1} scores are close to the quality threshold, but q^{f1} better captures the class imbalance problem. For the datasets derived from Spambase, we expect to obtain a good quality level, but the metric always returns medium. We always measure good

performance (accuracy and f1-score) with the first part of the metric (q_1^{acc}, q_1^{f1}). But, we observe that the second part (q_2^{acc}, q_2^{f1}) presented in Table 6 also captures the fact that small injections of errors (mostly missing values) in these datasets imply a high reduction of performance. This is indeed problematic, especially if the datasets are used for continuous machine learning. The 7 characteristics we have considered to manually evaluate data quality are not sufficient to detect this problem. So, we can consider that the results provided by our metric are correct.

Table 6: Percentages of variations of accuracy and f1-score for a set of deteriorated versions of the dataset Spambase (forced class imbalance) in response to the injection of 5% of missing values (M), outliers (O), and fuzzing (F).

Spambase	0%			5%			10%			30%			35%			50%		
	M	O	F	M	O	F	M	O	F	M	O	F	M	O	F	M	O	F
Variation of accuracy	6.8%	2.3%	0.3%	8%	2.6%	0.5%	10.7%	3.1%	0.2%	8.4%	2.9%	0.1%	8.7%	2.4%	0.1%	8%	3.5%	0.7%
Variation of f1-score	6.9%	2.4%	0.3%	8%	2.7%	0.5%	11.6%	3.2%	0.2%	9.4%	2.9%	0%	9%	2.4%	0.1%	8.8%	3.5%	0.7%

To summarize for $C1$, if both q^{acc} and q^{f1} are studied to evaluate data quality, these experiments show that DQ is generally effective in estimating the correct quality levels. If q^{acc} or q^{f1} is used separately, we obtain unexpected quality levels either on account of uncertainty (measures near thresholds) or because q^{f1} better detects class imbalance problems. Consequently, both q^{acc} and q^{f1} should be used to evaluate which measure is the most appropriate with regard to the specific objectives and characteristics of the classification.

In these experiments, we have shown that the aggregation of q^{acc} and q^{f1} into one value is not an easy task as both measures capture their own characteristics related to model performance. $max(q^{acc}, q^{f1})$ or the mean of both scores might be simple ways for getting a single data quality value, but these aggregations might lead to wrong interpretations, especially if the scores are close to quality level thresholds. If there is no strict time constraint, it seems more relevant to study both q^{acc} q^{f1} and to perform more analysis.

4.3 C2 Relevance Of *DQ* Without Trusted Test Set

Setup: to investigate C2, we computed q^{acc} and q^{f1} for the five datasets Spambase, Heart Disease, Abalone, Dry Beans, and Statlog, along with their variants, by using a trusted test set or by applying Algorithm 1, i.e., by uniformly sampling test and training sets 30 times. We then measured the differences $\Delta q^{\Phi}_{e,p}(D) := q^{\Phi}_{e,p}(D_{trusted}) - q^{\Phi}_{e,p}(D_{untrusted})$, which express the quality differences computed for a dataset D deteriorated with p percents of error e with or without trusted test set.

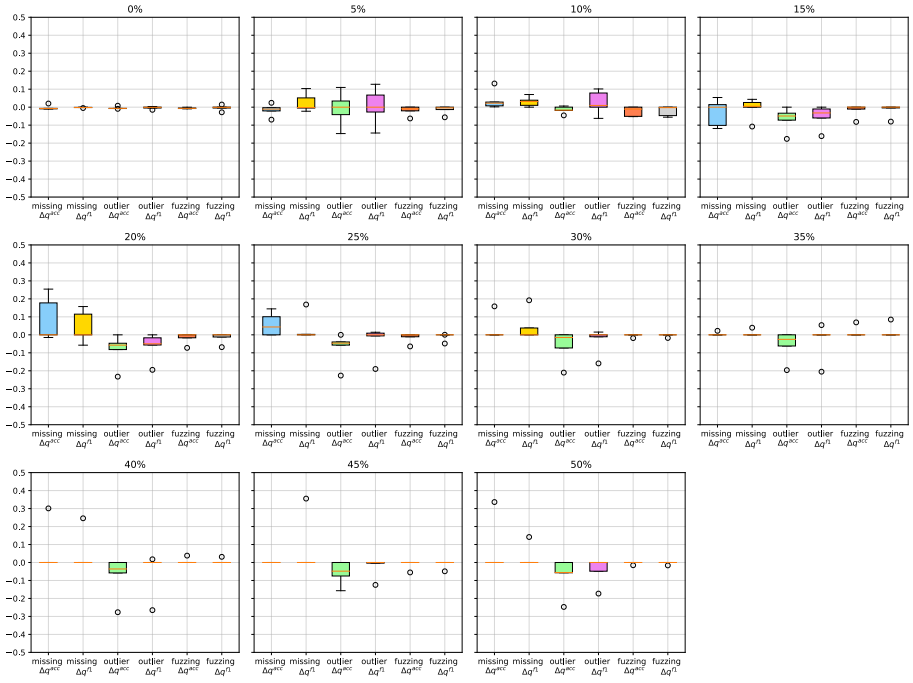


Fig. 6: Distributions of $\Delta q^{a}_{e,p}(D)$ and $\Delta q^{f1}_{e,p}(D)$ detailed for every 5% increment of missing values, outliers and fuzzing up until 50%.

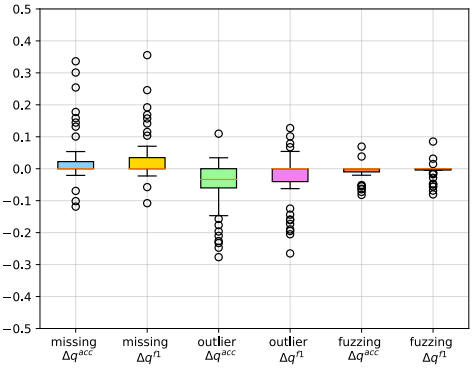


Fig. 7: Summary of the distributions of $\Delta q^{acc}_{e,p}(D)$ and $\Delta q^{f1}_{e,p}(D)$ for 0% to 50% of missing values, outliers and fuzzing.

Results: Figure 6 shows the distributions of $\Delta q^{acc}_{e,p}$ and $\Delta q^{f1}_{e,p}$ for the datasets with uniform injections of missing values, outliers, and fuzzing in 5% increments

from 0% up to 50%. Figure 7 summarizes these results with one boxplot for every error type and performance indicator (accuracy of f1-score). We observe that the means of Δq^{acc} and Δq^{f1} are close to 0 for the 3 errors. The boxplots illustrated in Figure 6 shows that we sometimes have slight differences though. The variations observed are primarily linked to the generation of few particular test sets through sampling, which yield lower or higher performance scores.

These observations, along with those performed in Section 2.3, show that the use of Algorithm 1 to compute DQ or the use of a trusted test set when available give similar quality scores. But, the obtained quality levels may be different if the q^{acc} or q^{f1} scores are close to quality level thresholds. It is manifest that, when available, relying on trusted test sets consistently yields superior results in assessing data quality.

4.4 C3 Comparison Of *DQ* With Quality Dimensions

We now compare DQ with three objective dimensions, Completeness, Class Imbalance, and Data Dimensionality, presented in [12,8,5,35,18]. We chose them because, as our metric DQ, they do not require any metadata to produce measurements. Besides, we have observed that these dimensions are often used by data scientists. These comparisons aim at checking whether DQ agree with the observations provided by these dimensions.

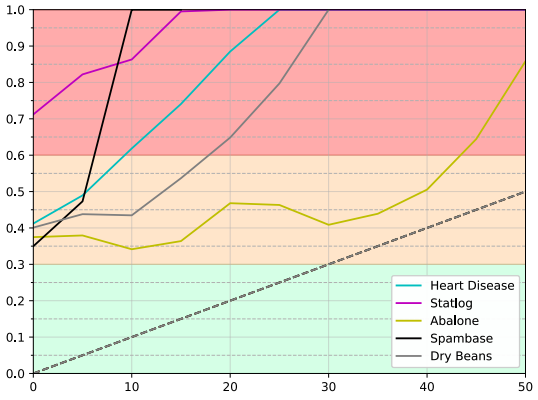


Fig. 8: $max(q^{acc}, q^{f1})$ when missing errors are injected in data (solid lines) and the completeness ratio (dashed lines).

Completeness

Setup: to evaluate Completeness, we used the ratio $\frac{\#missing\ values}{\#total\ values}$ [12]. Generally speaking, for a Completeness ratio equal to 0.3, the dataset quality is expected to be of medium or bad quality. For a ratio of 0.5, the dataset quality

is expected to be of bad quality. In order to compare DQ with this ratio, we computed them on the five datasets presented in Table 1 along with the 50 deteriorated datasets obtained after injecting 5 to 50% of missing values randomly, with a uniform distribution.

Results: Figure 8 illustrates the ratio with a dashed line and the scores $max(q^{acc}, q^{f1})$ with solid lines for each dataset, which are again represented in the x-axis. We chose to depict the maximum values for readability reasons. When the Completeness ratio is equal to 0.3 (30% of missing values injected in the datasets), we observe that all the datasets, except the one obtained from Abalone, are evaluated as bad by DQ. And at 0.5, all the datasets are rated as bad by DQ. Hence, DQ confirms with measurements the general expectations of the dimension, but DQ returns precise results and interpretations for each dataset. Hence, DQ should be used instead of this ratio. It is worth noting that we already eventuated the correctness of the DQ measurements for these datasets with C1.

Class Imbalance

Setup: we considered the ratio $\frac{\#samples(minority\ class)}{\#samples(majority\ class)}$ [18] to measure the Class Imbalance dimension. Generally speaking, a dataset having a ratio 30/70 is expected to be of medium or bad quality. With a ratio 10/90 or lower, a dataset is expected to be of bad quality and should be rejected. We computed this ratio and DQ for the five datasets presented in Table 1 and imbalanced datasets incrementally obtained by deleting samples in one class to obtain these imbalance ratios: $\frac{50}{50}$ (balanced classes, this corresponds to the original datasets), $\frac{40}{60}$, $\frac{30}{70}$, $\frac{20}{80}$, $\frac{10}{90}$, and $\frac{5}{95}$.

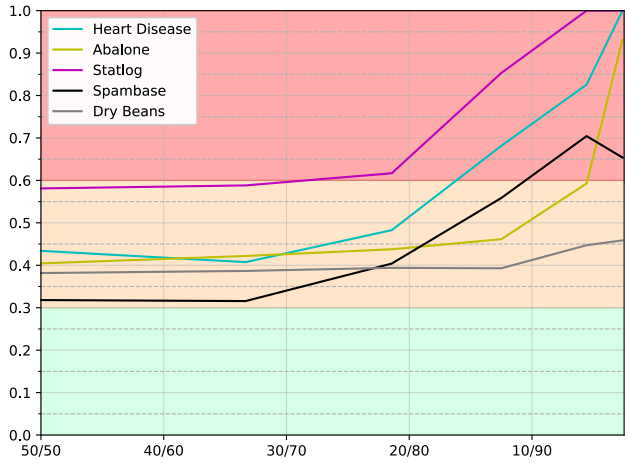


Fig. 9: $max(q^{acc}, q^{f1})$ on datasets having an imbalance ratio given by the x-axis.

Result: Figure 9 illustrates the $max(q^{acc}, q^{f1})$ scores for each dataset represented on the x-axis. We observe that the curves $max(q^{acc}, q^{f1})$ increase with the decrease of the class imbalance ratio. Therefore DQ captures the quality issues caused by class imbalance. But, DQ and our quality interpretations do not always meet the expected qualities suggested by the ratio. For instance, at 10/90, Dry Beans and Abalone are of medium quality whereas it is expected to have datasets of bad quality (we showed in C1, that the results given by DQ are correct for these datasets). These differences come from the fact that the Class Imbalance dimension is not sufficient to conclude on data quality. We can have a strong imbalance between a majority and a minority class but enough samples in these classes to get good classification results. This is the case for Dry Bean. However, it is worth noting that DQ is an objective metric. It cannot detect bias involved by imbalanced classes such as a lack of fairness.

Data Dimensionality

Setup: we formulate Data Dimensionality with the ratio $\frac{\#Features}{\#MaxFeatures}$. We measured DQ and this ratio for the 5 datasets of Table 1 along with 42 datasets derived from the previous ones as follows: we started by sorting the features of each dataset by importance using the random forest classifier. We reduced each dataset to its most important feature, then we added the remaining features incrementally, in order of importance, to build new datasets until all the features were added back.

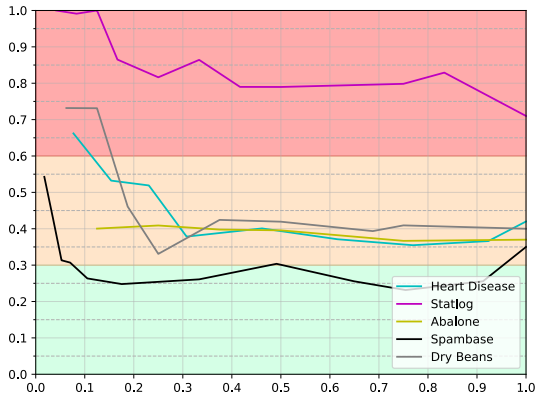


Fig. 10: $max(q^{acc}, q^{f1})$ as a function of $\frac{\#Features}{\#MaxFeatures}$ when we reduce Data Dimensionality by removing features.

Results: Figure 10 depicts the scores $max(q^{acc}, q^{f1})$ with the x-axis expressing the datasets having a ratio $\frac{\#Features}{\#MaxFeatures}$ between 0 to 1. As the initial datasets have different numbers of features ($\#MaxFeatures$), the curves have different starting points. We observe that $max(q^{acc}, q^{f1})$ generally decreases as

the number of features increases, which confirms what we expected, i.e. data quality should increase when more features are included. However, these experiments also confirm that Data Dimensionality cannot be used alone to measure data quality. Indeed, we can observe with DQ the fact that datasets with more features do not necessarily have better quality than datasets with less features. For instance, Spambase with all its features (ratio at 1) is of medium quality whereas Spambase with less features (ratio at 0.7) is of good quality. This can be explained by the fact that Spambase with all its features include useless features for classification that can hinder the convergence of the classifiers. This kind of observation cannot be done with the ratio. Again, DQ seems more relevant to measure data quality than this dimension.

These comparisons between objective dimensions and DQ suggest concluding that our metric produces correct and more precise data quality measurements than the values returned by the ratios considered for studying Completeness, Class Imbalance, or Data Dimensionality.

4.5 C4 Performance Of Algorithm 1

Setup: we investigated how Algorithm 1 scales with the dataset sizes. We measured the computational time for 28 datasets derived from Iris and Adult with a total number of samples ranging from 150 for Iris to 200 for Adult, and up to 20000 for both of them. In order to limit the influence of a particularly good or bad hyper-parameter tuning, we opted to use the default hyper-parameter settings in scikit-learn implementations for the models in M. Execution times were measured for a parallel computation of DQ with 8 CPUs on a laptop with standard configuration: 2.5GHz, Intel Core i7 4-core, 16Go RAM.

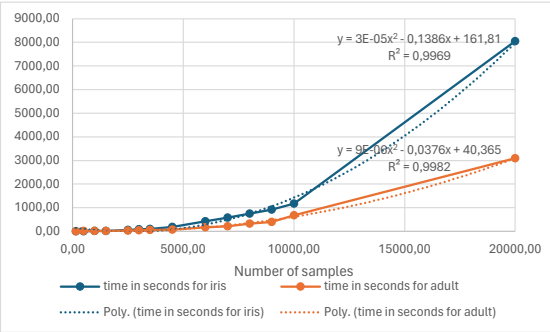


Fig. 11: Variation of the computational time (in seconds) as a function of dataset sizes

Results: Figure 11 depicts the computational time in seconds of our algorithm w.r.t. dataset size. We first observe that the curves are quite different: the

Computational time is generally tied to the complexity of the dataset structure, and we believe that this observed difference is influenced by the structures of Iris and Adult (Iris has 4 attributes and 3 classes and Adult has 10 attributes and 2 classes). At worst, with Adult, we observe that Algorithm 1 requires almost 2 hours and a half to compute DQ. Furthermore, both curves exhibit a quadratic trend, which indicates that our algorithm does not scale well. This is an expected issue inherent in our approach. The larger the dataset or the more complex the dataset structure, the longer it takes to compute DQ. Furthermore, if the set M of classification models is completed or if more complex models are used, the computation time might increase drastically. This issue needs to be investigated in future work.

4.6 Threats To Validity

In order to ensure the validity of the experimental results, we identified and addressed 8 possible threats: 4 internal and 4 external. The first factor, which may threaten the internal validity, concerns the implementation of classification models in M. To mitigate this, we used scikit-learn [34], a widely used library. The second threat is related to the hyperparameterization of these classification models. To address this threat, we used a grid search to set the hyper-parameters for classification models on all datasets without any deterioration and then used them for the rest of the experiments. The third and fourth threats are related to the number of datasets used to study the impact of not having trusted test set on measurements and those used to set the interpretation thresholds for DQ. To address these threats, we studied these aspects with three different datasets and their 175 variants. Besides, we evaluated these aspects with other datasets. But further datasets could be considered to confirm our results.

Regarding the external threats, the first one concerns the choice of the evaluation datasets. To ensure that our results are not biased, we chose freely available datasets and widely used in the literature. We also selected datasets that cover various applications and ranges of dimensions. The second threat is the choice of classification models. To mitigate this, we selected a wide range of classification approaches, from simple regressions to neuronal networks. The third threat is related to the generation of deteriorated datasets. To address this, we decided to generate them by injecting errors randomly, with a uniform distribution, in order to limit as much as possible the creation of additional bias. Finally, the last threat is the error types we considered to generate the deteriorated datasets. We considered three common error types along with the class imbalance problem. However, datasets may also contain other types of errors or combinations. Consequently, our results and conclusions are limited by this choice, and further experiments will be necessary to address these limitations.

5 Conclusion

We have introduced a novel data quality metric specifically designed for numeric datasets in classification tasks. The proposed metric is based on the correlated

evolution between the classification performance and data deterioration. Extensive experiments and illustrations demonstrate its primary benefits: Being model-independent and not requiring external data or expert knowledge. Moreover, its inherent flexibility allows easy adaptation for diverse real-life scenarios. To make the proposed metric easy to interpret, we have provided a comprehensive interpretation of its scores and illustrated key assessment levels. Based on extensive experiments and comparisons with some objective state-of-the-art quality dimensions, we have shown that the proposed metric has succeeded in characterizing data quality for different types of datasets including use cases of good, medium, and bad quality. Data, results, and the code are available online via the link [25].

For Future works, several aspects require further investigation and improvement. Initially, we aim to further explore the adaptability and versatility of DQ by expanding its evaluation with additional datasets. We indeed considered numeric datasets, but there is a wide variety of numeric data, and our evaluation does not include them all. For instance, incorporating datasets of images, audio data, geospatial data, and other sub-types of numeric data that can be very specific to some applications, such as medical imaging, could be a valuable addition. Another future work to consider when computing DQ is the measurement of performance with test sets. In our current DQ implementation, we use a trusted test set if available or compute DQ as a mean over 30 random resamplings of training and test sets otherwise. While resampling is a common approach, it may not always be optimal, and various alternative techniques have been proposed (see [40]). Mutation testing [29] is another approach to improve the test set quality. However this approach is time-consuming and is not generalized for any classification model. Exploring the extension of our metric to other types of machine learning, such as regression or clustering, is also worth considering. Further studies and evaluations are here necessary to ascertain the compatibility of DQ with other performance indicators associated with these tasks and to refine its interpretation.

6 Annex

Table 7: Data quality when 10% and 15% of missing values are injected in data

Dataset	Number of classes	Number of samples	Number of attributes	$CImb(D)$	Values at 10% of deterioration $Comp(D) = 0.9$			Values at 15% of deterioration $Comp(D) = 0.85$		
					accuracy	f1-score	Estimated data quality	accuracy	f1-score	Estimated data quality
Heart Disease	2	297	13	0.08	0.72	0.69	medium	0.66	0.63	bad
Statlog	2	959	23	0.43	0.7	0.57	bad	0.61	0.5	bad
Abalone	2	4 177	8	0.33	0.85	0.83	medium	0.85	0.82	medium
Spambase	2	4 601	57	0.21	0.51	0.41	bad	nan	nan	bad
Dry Beans	7	13 611	16	0.35	0.67	0.63	medium	0.6	0.54	medium

Table 8: Data quality when 10% and 15% of outliers are injected in data

Dataset	Number of classes	Number of samples	Number of attributes	$CImb(D)$	$Comp(D)$	Values at 10% of deterioration			Values at 15% of deterioration		
						accuracy	f1-score	Estimated data quality	accuracy	f1-score	Estimated data quality
Heart Disease	2	297	13	0.08	1	0.75	0.74	medium	0.72	0.71	medium
Statlog	2	959	23	0.43	1	0.72	0.57	bad	0.71	0.53	bad
Abalone	2	4 177	8	0.33	1	0.81	0.78	medium	0.8	0.77	medium
Spambase	2	4 601	57	0.21	1	0.8	0.79	medium	0.77	0.76	medium
Dry Beans	7	13 611	16	0.35	1	0.59	0.57	medium	0.56	0.53	medium

Table 9: Data quality when 10% and 15% of one class is deleted (we start with balanced classes)

Dataset	Number of classes	Number of attributes	$Comp(D)$	Values at 10% of deterioration					Values at 15% of deterioration				
				$CImb(D)$	Number of samples	accuracy	f1-score	Estimated data quality	$CImb(D)$	Number of samples	accuracy	f1-score	Estimated data quality
Heart Disease	2	13	1	0.04	263	0.81	0.8	medium	0.07	257	0.77	0.76	medium
Statlog	2	23	1	0.04	528	0.73	0.72	medium	0.06	517	0.73	0.72	medium
Abalone	2	8	1	0.04	2 701	0.79	0.79	medium	0.06	2 645	0.79	0.79	medium
Spambase	2	57	1	0.04	3 481	0.92	0.92	good	0.06	3 408	0.9	0.9	good
Dry Beans	7	16	1	0.02	3 612	0.69	0.65	medium	0.03	3 591	0.7	0.67	medium

References

1. Adult. UCI Machine Learning Repository (1996), DOI: 10.24432/C5XW20
2. Abdelaal, M., Hammacher, C., Schoening, H.: Rein: A comprehensive benchmark framework for data cleaning methods in ml pipelines. arXiv preprint arXiv:2302.04702 (2023)
3. Askham, N., Cook, D., Doyle, M., Fereday, H., Gibson, M., Landbeck, U., Lee, R., Maynard, C., Palmer, G., Schwarzenbach, J.: The six primary dimensions for data quality assessment. DAMA UK working group pp. 432–435 (2013)
4. Ataccama: Ataccamaone. https://www.ataccama.com/platform (2023)
5. Batini, C., Cappiello, C., Francalanci, C., Maurino, A.: Methodologies for data quality assessment and improvement. ACM computing surveys (CSUR) **41**(3), 1–52 (2009)
6. Batini, C., Scannapieco, M., et al.: Data and information quality. Cham, Switzerland: Springer International Publishing (2016)
7. Bors, C., Gschwandtner, T., Kriglstein, S., Miksch, S., Pohl, M.: Visual interactive creation, customization, and analysis of data quality metrics. In: Journal of Data and Information Quality (JDIQ) ACM (2018)
8. Cichy, C., Rass, S.: An overview of data quality frameworks. IEEE Access **7**, 24634–24648 (2019)
9. Cortes, C., Jackel, L.D., Chiang, W.P.: Limits on learning machine accuracy imposed by data quality. Advances in Neural Information Processing Systems **7** (1994)

Table 10: Data quality when 30% and 35% of missing values are injected in data

Dataset	Number of classes	Number of samples	Number of attributes	$CImb(D)$	Values at 30% of deterioration $Comp(D) = 0.7$			Values at 35% of deterioration $Comp(D) = 0.65$		
					accuracy	f1-score	Estimated data quality	accuracy	f1-score	Estimated data quality
Heart Disease	2	297	13	0.08	0.52	0.52	bad	nan	nan	bad
Statlog	2	959	23	0.43	nan	nan	bad	nan	nan	bad
Abalone	2	4 177	8	0.33	0.82	0.8	medium	0.8	0.78	medium
Spambase	2	4 601	57	0.21	nan	nan	bad	nan	nan	bad
Dry Beans	7	13 611	16	0.35	nan	nan	bad	nan	nan	bad

Table 11: Data quality when 30% and 35% of outliers are injected in data

Dataset	Number of classes	Number of samples	Number of attributes	$CImb(D)$	$Comp(D)$	Values at 30% of deterioration			Values at 35% of deterioration		
						accuracy	f1-score	Estimated data quality	accuracy	f1-score	Estimated data quality
Heart Disease	2	297	13	0.08	1	0.65	0.64	bad	0.66	0.64	bad
Statlog	2	959	23	0.43	1	0.7	0.5	bad	0.7	0.48	bad
Abalone	2	4 177	8	0.33	1	0.77	0.72	medium	0.76	0.7	medium
Spambase	2	4 601	57	0.21	1	0.73	0.69	medium	0.72	0.67	medium
Dry Beans	7	13 611	16	0.35	1	0.5	0.47	medium	0.49	0.46	medium

Table 12: Data quality when 30% and 35% of one class is deleted (we start with balanced classes)

Dataset	Number of classes	Number of attributes	$Comp(D)$	Values at 30% of deterioration					Values at 35% of deterioration				
				$CImb(D)$	Number of samples	accuracy	f1-score	Estimated data quality	$CImb(D)$	Number of samples	accuracy	f1-score	Estimated data quality
Heart Disease	2	13	1	0.14	241	0.8	0.8	medium	0.17	235	0.76	0.76	medium
Statlog	2	23	1	0.14	484	0.72	0.71	medium	0.16	473	0.71	0.7	medium
Abalone	2	8	1	0.14	2 476	0.79	0.79	medium	0.16	2 420	0.79	0.78	medium
Spambase	2	57	1	0.14	3 191	0.9	0.9	good	0.16	3 118	0.9	0.9	good
Dry Beans	7	16	1	0.06	3 528	0.72	0.7	medium	0.07	3 508	0.7	0.68	medium

10. DataCleaner: Datacleaner. https://datacleaner.github.io/ (2023)
11. Datamartist: Datamartist. http://www.datamartist.com/ (2023)
12. Ehrlinger, L., Wöß, W.: A survey of data quality measurement and monitoring tools. Frontiers in big data p. 28 (2022)
13. Experian: User manual version 5.9. https://www.edq.com/globalassets/documentation/pandora/pandora/manual/590.pdf (2023)
14. Fisher, R.: Iris. UCI Machine Learning Repository (1988), DOI: 10.24432/C56C76
15. Foundation, A.: Apache griffin user guide. https://github.com/apache/griffin/blob/master/griffin-doc/ui/user-guide.md (2023)
16. Gudivada, V., Apon, A., Ding, J.: Data quality considerations for big data and machine learning: Going beyond data cleaning and transformations. International Journal on Advances in Software **10**(1), 1–20 (2017)
17. Haegmans, T., Snoeck, M., Lemahieu, W.: Towards a precise definition of data accuracy and a justification for its measure. Proceedings of the International Conference on Information Quality p. 16 (2016), mIT Information Quality (MITIQ) Program
18. Hamid, M.H.A., Yusoff, M., Mohamed, A.: Survey on highly imbalanced multi-class data. International Journal of Advanced Computer Science and Applications **13**(2) (2022)
19. Heinrich, B., Klier, M.: A novel data quality metric for timeliness considering supplemental data. In: Information systems in a globalising world : challenges, ethics and practices; 17th European Conference on Information Systems. pp. 2701–2713 (2009)
20. Hinrichs, H.: Datenqualitätsmanagement in data warehouse-systemen. Ph.D. thesis, Universität Oldenburg (2002)
21. IBM: Ibm data quality for ai api. https://developer.ibm.com/apis/catalog/dataquality4ai--data-quality-for-ai/Introduction (2023)
22. Informatica: What is data quality? https://www.informatica.com/resources/articles/what-is-data-quality.html (2023)
23. InfoZoom: Infozoom & izdq. https://www.infozoom.com/en/products/infozoom-data-quality/ (2023)
24. Jouseau, R., Salva, S., Samir, C.: On Studying the Effect of Data Quality on Classification Performances. In: Intelligent Data Engineering and Automated Learning - IDEAL 2022. Lecture Notes in Computer Science, vol. 13756, pp. 82–93. Springer International Publishing, Manchester, United Kingdom (Nov 2022). https://doi.org/10.1007/978-3-031-21753-1_9, https://hal.uca.fr/hal-03938077
25. Jouseau, R., Salva, S., Samir, C.: Additional resources for the reproducibility of the experiment. https://gitlab.com/roxane.jouseau/measuring-data-quality-for-classification-tasks (2023)
26. Jouseau, R., Salva, S., Samir, C.: A novel metric for measuring data quality in classification applications. In: Rocha, A.P., Steels, L., van den Herik, H.J. (eds.) Proceedings of the 16th International Conference on Agents and Artificial Intelligence, ICAART 2024, Volume 3, Rome, Italy, February 24-26, 2024. pp. 141–148. SCITEPRESS (2024). https://doi.org/10.5220/0012311500003636, https://doi.org/10.5220/0012311500003636
27. Lettner, C., Stumptner, R., Fragner, W., Rauchenzauner, F., Ehrlinger, L.: Daql 2.0: Measure data quality based on entity models. Procedia Computer Science **180**, 772–777 (2021)
28. Li, P., Rao, X., Blase, J., Zhang, Y., Chu, X., Zhang, C.: Cleanml: A study for evaluating the impact of data cleaning on ml classification tasks. 36th IEEE International Conference on Data Engineering (ICDE 2020)(virtual) (2021)

29. Ma, L., Zhang, F., Sun, J., Xue, M., Li, B., Juefei-Xu, F., Xie, C., Li, L., Liu, Y., Zhao, J., Wang, Y.: Deepmutation: Mutation testing of deep learning systems. arXiv:1805.05206 [cs] (2018)
30. Markelle Kelly, Rachel Longjohn, K.N.: The uci machine learning repository, https://archive.ics.uci.edu
31. Moore, S.: How to create a business case for data quality improvement. Gartner Research (2018)
32. Neutatz, F., Chen, B., Alkhatib, Y., Ye, J., Abedjan, Z.: Data cleaning and automl: Would an optimizer choose to clean? Datenbank-Spektrum **22**(2), 121–130 (2022)
33. OpenRefine: Openrefine. https://github.com/OpenRefine/OpenRefine (2023)
34. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: Machine learning in Python. Journal of Machine Learning Research **12**, 2825–2830 (2011)
35. Pipino, L.L., Lee, Y.W., Wang, R.Y.: Data quality assessment. Communications of the ACM **45**(4), 211–218 (2002)
36. Ridzuan, F., Wan Zainon, W.M.N.: A review on data cleansing methods for big data. Procedia Computer Science **161**, 731–738 (2019). https://doi.org/https://doi.org/10.1016/j.procs.2019.11.177, https://www.sciencedirect.com/science/article/pii/S1877050919318885, the Fifth Information Systems International Conference, 23-24 July 2019, Surabaya, Indonesia
37. Rolland, A.: Mobydq. https://ubisoft.github.io/mobydq (2023)
38. SAS: Dataflux data management studio 2.7: User guide. http://support.sas.com/documentation/onlinedoc/dfdmstudio/2.7/dmpdmsug/dfUnity.html (2023)
39. Sebastian-Coleman, L.: Measuring data quality for ongoing improvement: a data quality assessment framework. Newnes (2012)
40. Sherin, S., Iqbal, M.Z., et al.: A systematic mapping study on testing of machine learning programs. arXiv preprint arXiv:1907.09427 (2019)
41. Talend: Talend open studio for data quality – user guide 7.0.1m2. http://download-mirror1.talend.com/top/user-guide-download/V552/TalendOpenStudio_DQ_UG_5.5.2_EN.pdf (2023)
42. Wang, R.Y., Strong, D.M.: Beyond accuracy: What data quality means to data consumers. Journal of management information systems **12**(4), 5–33 (1996)
43. William, W., W., S., O., M.: Breast Cancer Wisconsin (Diagnostic). UCI Machine Learning Repository (1995), DOI: 10.24432/C5DW2B