The ABC of data

The ABC of Data: A Classifying Framework for Data Readiness

Laurens A. Castelijns[1,2,3], Yuri Maas[1,2,3], and Joaquin Vanschoren[1]

[1] Faculty of Mathematics & Computer Science
Eindhoven University of Technology, Eindhoven, Netherlands
[2] School of Law, Tilburg University, Tilburg, Netherlands
[3] Jheronimus Academy of Data Science, 's-Hertogenbosch, Netherlands

Abstract. In order to (semi)automate data cleaning and preprocessing, we need a clear and measurable definition of data quality. Data readiness levels have been proposed to fit this need, but they require a more detailed and measurable definition than is given in prior works. We present a practical framework focused on machine learning that encapsulates data cleaning and (pre)processing procedures. In our framework, datasets are classified within bands, and each band introduces more fine-grained terminology and processing steps. Scores are assigned to each step, resulting in a data quality score. This allows teams of people, as well as automated processes, to track and reason about the cleaning process, and communicate the current status and deficiencies in a more structured, well-documented manner.

Keywords: Data Quality — Data Readiness Levels — Data Cleaning — Preprocessing — Automated Data Science

1 Introduction

Data is the new oil. It is valuable, but if unrefined it cannot really be used. It has to be changed into gas, plastic, chemicals, etc to create a valuable entity that drives profitable activity; so must data be broken down, analyzed for it to have value.

The popular metaphor between data and oil is credited to the British mathematician Clive Humby in 2006. There are many ways in which his analogy might be broken down but Dr. Humby here points out an incontestable truth: Data needs processing.

We pose that data (pre)processing aims to increase *data quality*, and present a practical framework that encapsulates a range of data processing steps to achieve this. Inspired by the concepts introduced by Neil D. Lawrence in his position paper on *Data Readiness Levels* [13], it examines and structures the technical challenges that, when solved, increase data quality. The resulting framework is used as the theoretical foundation of the software package PyWash [3], a collection of tools used to clean and process datasets to increase their quality.

Raw data usually suffers from a wide range of issues, such as duplicated records, missing values, outliers, typos, and many other issues that weaken the quality of the data and hinder advanced analysis. This results in machine learning systems learning the wrong things, decreasing their accuracy and making them unreliable at best and plain wrong at worst. Data cleaning is, therefore, an essential task. Data cleaning is often an iterative process that is tailored to the needs and wants of a specific analysis task. Krishnan et. al (2015) conducted a survey that expresses the need for a streamlined data cleaning framework. The question "How do you determine whether the data is sufficiently clean to trust the analysis?" made clear that most of the respondents had no rigorous validation of their data cleaning. In response to this survey the same authors created ActiveClean [12], which describes an iterative cleaning process that selects and cleans some records. After this cleaning, it measures the performance of the dataset on the main analysis to then select and evaluate if more cleaning is necessary. The iterative nature of data cleaning paired with the absence of an evaluation methodology is alarming. Alternating between cleaning data and analyzing data, and using these analysis results to guide the subsequent data cleaning procedures can result in overfitting. Data cleaning procedures are generally under-reported because it is such a 'dirty' process. Often there is no log maintained of data cleaning operations executed whilst these operations can introduce bias into the dataset [11].

Since then, some attempts have been made to set up a data quality framework. "InfoQ" breaks down data analysis into 4 distinct components: analysis goal, available data, utility measure, and data analysis method. The information quality is then assessed using eight dimensions, such as data structure (explained as the type of data and data characteristics) and temporal relevance. The quality of each of these eight dimensions is then assessed separately, often using a rating on a Likert scale. There are multiple approaches to then compute an overall InfoQ score by properly combining this set of eight assessments [7]. For example, Ron Kenett and Marco Reis applied InfoQ to the Chemical Processing Industry and proposed an assessment strategy in which each dimension is weighted to reflect the distinct focal points in different analysis goals [16]. A limitation of InfoQ is that a Likert scale abstracts away from the actual operations that have to be performed to increase data quality. If you were told that a dataset obtained an InfoQ score of 77% it is not clear what kind of deficiencies are present. After sharing the individual Likert scale scores for each dimension it is still unknown what exactly can be done to improve the score (and in what order). Neil D. Lawrence (2017) recognized the overall lack of terminology in discussions about data quality and proposed an initial set of descriptors for data readiness. The proposal is to split data readiness into three distinct *bands*. The bands are represented by the letters: A, B, and C. Each band contains sub levels: A1 is data of the highest quality and C4 would be data of the worst quality [13]. However, the author refrains from elaborating the bands in greater detail and therefore the bands remain vague. In this paper, we propose one way to further extend, detail and quantify these bands.

2 The Framework

We introduce a framework which streamlines and describes the data cleaning process. The framework splits the cleaning process into multiple distinct categories we likewise call *bands* and it analyzes the dataset to determine in which band it currently is. Datasets that are in a certain band may possess one or more deficiencies which are specified for that band and will negatively influence any analysis, such as machine learning, performed on the dataset. Thus, in order for a dataset to be classified as a higher tier band and be deemed *cleaner*, the issues from the current band have to be resolved.

The bands introduce steps and terminology in the cleaning process that are easy to follow for most practitioners. Teams will be able to communicate, argue and customize the cleaning process to better fit their needs in a structured, potentially well-documented process.

This new way of thinking about data cleaning as a step-by-step process and a standalone part of the whole data process will hopefully save people from rushing through data cleaning, and provide them with useful data quality metrics rather than purely optimizing a final model quality score (e.g. model accuracy). Moreover, it will also help to increasingly automate the process while alleviating overfitting.

2.1 Data Bands

The framework consists of the following bands: C, B, A, AA, and AAA. These represent the different stages of usability that datasets can be in during the process.

Band C (Conceive) refers to the stage that the data is still being ingested. If there is information about the dataset, it comes from the data collection phase and how the data was collected. The data has not yet been introduced to a programming environment or tool in a way that allows operations to be performed on the dataset. The possible analyses to be performed on the dataset in order to gain value from the data possibly haven't been conceived yet, as this can often only be determined after inspecting the data itself.

Band B (Believe) refers to the stage in which the data is loaded into an environment that allows cleaning operations. However, the *correctness* of the data is not fully assessed yet, and there may be errors or deficiencies that would invalidate further analysis. Therefore, analyses performed in this stage are often more cursory and exploratory, such as a exploratory data analysis with visualization methods to ascertain the correctness of the data. Skipping these checks might lead to errors or 'wrong' results and conclusions.

In **band A (Analyze)**, the data is ready for deeper analysis. However, even if there are no more factual errors in the data, the quality of an analysis or machine learning model is greatly influenced by how the data is represented. For instance, operations such as feature selection and normalization can greatly increase the accuracy of machine learning models. Hence, these operations need

to be performed before arriving at accurate and adequate machine learning models or analyses. In many cases, these operations can already be automated to a significant degree.

In **band AA (Allow Analysis)**, we consider the context in which the dataset is allowed to be used. Operations in this band detect, quantify, and potentially address any legal, moral or social issues with the dataset, since the consequences of using illegal, immoral or biased datasets can be enormous. Hence, this band is about verifying whether analysis can be applied without (legal) penalties or negative social impact. One may argue that legal and moral implications are not part of data cleaning, but rather distinct parts of the data process. However, we argue that readiness is about learning the ins and outs of your dataset and detecting and solving any potential problems that may occur when analyzing and using a dataset.

Band AAA is the terminus of our framework. Getting into AAA would mean that the dataset is clean. The data is self-contained and no further input is needed from the people that collected or created the data.

2.2 Quality Scores

A dataset has a score between 0 and 1 for each band of our framework, so a dataset can have score 0.9 for band C, 0.8 for band B, 0.10 for A, 0.20 for AA and possibly 0 for band AAA. Datasets start with initial band score values of 0 for every band, as we generally do not know (for certain) which issues the dataset is suffering from that could potentially jeopardize machine learning methods. The dataset is classified in band C at this stage. We then proceed to check and solve all issues from band C. Each band deficiency that is solved or non-existent contributes to the band score of the dataset. Partially checked or solved deficiencies can grant partial weight scores. A dataset will move to the next band only when it has surpassed a certain *threshold* score. This also means that a dataset cannot get a band label of A or above when it has a B.60 score, even if the dataset fulfills all band A requirements.

The threshold scores can be determined by the framework users to determine how thoroughly the dataset has to be cleaned before it is able to proceed to further bands. We have set the default threshold for all bands on 0.85 to allow a dataset to advance while it's not totally perfect, since striving for a perfect dataset may not be achievable or cost-effective in general. The dataset might not be entirely clean when the thresholds are less than 1, as a dataset could advance to the next band (including band AAA) while not every issue has been checked or fixed yet. That said, the thresholds cannot be set too low (e.g., < 0.65) as datasets wouldn't be checked properly, which could seriously impact machine learning methods and dataset usability, causing errors and false predictions or estimates.

This terminology makes it easier to track and communicate the cleaning progress to others. This is because others will be able to understand what has to be done when a dataset is currently in band B, but might not know what to do next when only given a list of completed cleaning methods.

3 The different levels of data readiness

A dataset will be ready for certain operations to be performed on it depending on its band. The bands consist of several weighted dataset deficiencies which reflect what are currently the most important deficiencies that need to be addressed in the current band. An overview of the bands and their deficiencies can be found in table 1. A description of the bands, the functionality they unlock and their deficiencies are given below. The weights that we have given in table 1 are advisory weights for a generic dataset without a specific target analysis. In some scenarios, people may decide to use a different weighting. For example, medical applications may prioritize outlier detection, since detecting and investigating anomalies may have greater importance compared to other fields.

Band	W	Deficiency
C	40	Parseability
	25	Data storage
	15	Decoding
	10	Data Formats
	10	Disjoint Datasets
B	20	Column Types
	30	Missing Values
	20	Inconsistent Data Entries
	10	Duplicated Records
	20	Meaningful Values
A	20	Interpretable Values
	20	Feature Scaling
	20	Outlier Detection
	30	Feature Selection
	10	Coverage gap detection
AA	40	Legal Violations
	40	Security Risks
	20	Bias Detection
AAA		None

Table 1. The framework bands with weights and deficiencies

3.1 Band C: Conceive

Band C, and so too the framework, starts with having *access* to files or databases with the actual data. Data access has many problems of its own: datasets may be stored in a remote system with limited access or hidden in a large corporate ecosystem where few know the exact location of the desired dataset, thus human

interaction may be required before programmatic access is possible. Having access to the data is a prerequisite to even begin assessing its quality, hence the dataset's score will be 0 until access has been obtained. Such *hearsay data* [13] is therefore outside the scope of our framework.

That said, when we do have access, the data enters band C and tests have to be performed to see if the dataset is compatible with a programming environment or tool. Indeed, data files cannot provide value as is. Some procedures have to be followed before even basic analysis can be done. The aim in band C is to test for, and fix if necessary, the data deficiencies that are described below.

Parseability
There are many different file formats to store a dataset for long-term storage and transport, such as CSV, JSON and plain text. These formats specify how data can be loaded into a programming environment or analysis tool to perform operations. Therefore, making sure that a dataset can be loaded without errors receives a high weight in band C. This also includes the requirement of data access. You may know that a dataset exists, but technical or legal barriers might make it impossible to actually load and use the files.

Data Storage
The dataset needs to be stored in an effective and efficient manner relative to the operations that will have to be performed on the dataset. Getting the data in such a shape is often called *data wrangling*. The storage method does not have to be optimal, so there is no set limit to the runtime of the operations set by this paper. However, all operations of the subsequent bands must be executable. These operations aren't possible when the dataset isn't able to be stored in a way that allows these operations, thus checking how the data is stored is part of band C.

Decoding
The data has to be recognizable as data. This means that the data formatter should be able to use encoding styles that are known and understood by the environment that processes the dataset. The largest problem is that there are many different character encodings, and datasets can use any one of them. Common character encoding formats include ASCII [15], ISO 8859 and UTF-8 [18]. Luckily, automatic encoding detection has long been available [14].

Nevertheless, a system won't be able to use and find meaning in the data if the system is not able to distinguish or relate characters to each other. Thus datasets containing unknown formats cannot reliably be used to perform meaningful operations on. Which is why we classify any such datasets as being in band C.

Data Formats

Datasets are not always stored cleanly in a particular format. Human or technical problems can occur which might result in writing errors during the data collection phase. There are several different ways a data format may break. CSV files could change their separator halfway through the file, a JSON file misplaces a bracket or an ARFF file misses a categorical value. Mistakes happen and when they do, the parsing of data becomes difficult and could lead to unexpected outcomes. Therefore, a system has to check if a dataset has a consistent format and, even though it's not easy, should be able to fix most potential issues. This is different from being able to load the dataset since an incorrect format may not raise an error, but will change the structure of the dataset.

Disjoint datasets

Datasets can be divided up into multiple tables over several different files. Since they essentially are just one spread out dataset, any analysis should be performed on the entire dataset, rather than just one subset of it. Performing analysis on partial data can invite bias, and multiple analyses on the different dataset parts may introduce false positive errors and increase result variance.

3.2 Band B: Believe

In band B are datasets that are loaded into memory but still defective in some ways, which means that the data cannot be trusted at this moment. The data must be checked for trustworthiness and correctness of the data itself. After checking for these deficiencies and rectifying them, basic analytics can be used to explore the dataset.

Known and correct data types

Columns should have the correct data type (boolean, integer, float, date, categorical, ordinal, and string). A counterexample would be that a column is labeled as integer while it is effectively a non-ordinal encoding of categorical values (e.g. 1=blue, 2=green, 3=red) [17].

Missing values are identified and appropriately dealt with

Missing values are encoded as a variety of characters such as null, N/A, na, ?, and -1. The data points or features with missing data should be either removed or repaired (e.g. imputed) [1]. However, in some cases it does no harm to keep the missing values as long as they are properly identified.

Redundancy

We need to assess the degree to which there are duplicate records and columns.

Typos and inconsistent data entries
Imagine a column with colors which has red but also read. These values should be fixed or removed if the true value is unclear [9].

Meaningful values
If possible, variables should be expressed in a unit that is most suitable for machine learning. As a counterexample: a column with the height of people is expressed as a combination of feet and inches and encoded as a string. This can be useless for some machine learning models since no distance metric can be computed. This is also the point where clearly faulty data points -such as a name in a postcode field- should be removed [9].

3.3 Band A: Analyze

In band A are operations that further optimize and clean the data. The data is modified such that it is in a format that is properly suited for machine learning. Data that is not needed is filtered out to reduce overhead and increase accuracy.

Interpretable values
In is important that we know exactly what each feature (variable) in the data means. Maintaining a codebook [2] with a semantic description as possibly a unit for each feature is a good practice. For instance, if a column is named price but has no currency attached to it, it cannot be clearly interpreted. Knowing the right semantics also enables algorithmic transformation to a more convenient unit if needed [9]. Moreover, this also includes mappings for categorical values when they are encoded numerically, so that it is known what the numeric codes represent.

Feature Scaling
In feature scaling, you transform the data such that it is within a specific range. Some scaling methods such as normalization and standardization also change the distribution of the data. For some machine learning models, it is beneficial to scale the data. Neural networks, for example, are known to converge more quickly on normalized data.

Outlier Detection
As we have cleaned obvious faulty data points in band B, we are now able to search for naturally occurring data points that differ significantly from other observations. Outliers must be dealt with appropriately because not every machine learning model is robust to outliers [10].

Feature Selection

Some features may be redundant, for instance, a column may portray the same information as the column next to it. Removing the column will decrease training time and lower the risk of overfitting [6]. Dimensionality reduction techniques may also be used to represent high-dimensional data in fewer dimensions that facilitate modelling (but decrease interpretability).

Coverage gap detection

Data may have gaps such as spatial or temporal coverage gaps, For instance: sensor data was collected for 2 years, but because of a defect in the measurement equipment a part of the data is absent [9].

3.4 Band AA: Allow Analysis

We check the context in which the data is to be used in band AA. Data often originates from real people and is used to make decisions for real people. Thus the information of a dataset has to be placed in context to make sure that any analysis performed on the dataset will be permissible, usable and allowed. This is often not easily measurable, as this requires a lot of metadata from the dataset and from the context of possible analyses and how their results may be used. However, in most cases manual checks should be possible and should be done to ensure a dataset is allowed to be used in the real world.

Legal Violations

Laws and regulations can prevent the allowed usage of a dataset for analysis, even if the dataset was legally collected. This can happen when the data contains wrong values about the data subjects, or when the dataset is used outside the original scope. Information stored in datasets can also be harmful to the data subjects, even if the contents of the dataset are not leaked or stolen. That's why datasets that violate the GDPR [5] or other privacy regulations can result in large fines being imposed on the company or institution that is creating or using those datasets.

We deem these violations to be very important, both for any data processor and for all subjects within the data. That's why we propose to put a high weight on checking and solving any problems relating to this topic. Note that in many contexts, legal restrictions could inhibit anyone from even loading the data. In that case they would fall in band C (or pre-C).

Security Risks

Data security is a must when performing analyses on sensitive data such as personal data. Datasets containing sensitive information should be secure by design and security systems should already exist at the data collection stage. Consequently, the data should already be secure when entering the data cleaning phase. However, when this is not the case, the data cleaning phase should

prioritize the security of the data and transform and/or protect the data in such a way that any insecurities are prevented, either by encrypting the data or by securing the network the data is stored in. This also includes technical security procedures to work with the data [4].

A dataset is often secure because of the security systems around the dataset (e.g. firewalls), not because of the dataset itself. But it is essential that a dataset is protected, that's why we assigned such a high weight for checking the security of a dataset.

Bias detection

Bias can occur when the data collection samples neglect a portion of the entire population or when it over-samples on a specific portion. Analysis performed on datasets often separates the instances of the dataset into several groups to generalize them and suggest different actions for the different groups. This can become unethical (and illegal) when the groups are created based on discriminatory attributes (either directly or indirectly due to data collection bias). This is especially unethical when the results of the main analysis are used in decision-based applications, since such applications can and will discriminate. Therefore, datasets should be checked before any analysis, and any resulting models should only make predictions on cases that are sufficiently covered by their training data.

Techniques and frameworks exist to detect and eliminate dataset bias during training, and these should be an integral part of the cleaning process [8]. However, it is hard to eliminate *unknown* biases during the cleaning process.

3.5 Band AAA: A Clean Dataset

If a dataset has passed all bands, then it is considered as properly cleaned based on how the thresholds and weights were chosen. As mentioned before, dataset can advance to the next band while not every single issue has been completely solved. Thus a dataset might not be perfectly clean when in band AAA, but it is clean enough such that we can use it in most applications.

Band AAA does not contain any issues and doesn't add any functionality. It is rather an indicator that a dataset has completed the cleaning process and is ready to be effectively used in the determined main analysis.

4 Deployment

As mentioned in the introduction, this framework is used as the theoretical foundation of the practical tool *PyWash* [3]. PyWash is a python package that combines the described framework with a set of (semi)automatic data cleaning and preprocessing methods. The package will analyze datasets, assign band scores, and guide the user to the appropriate tools to import, clean, and export the dataset.

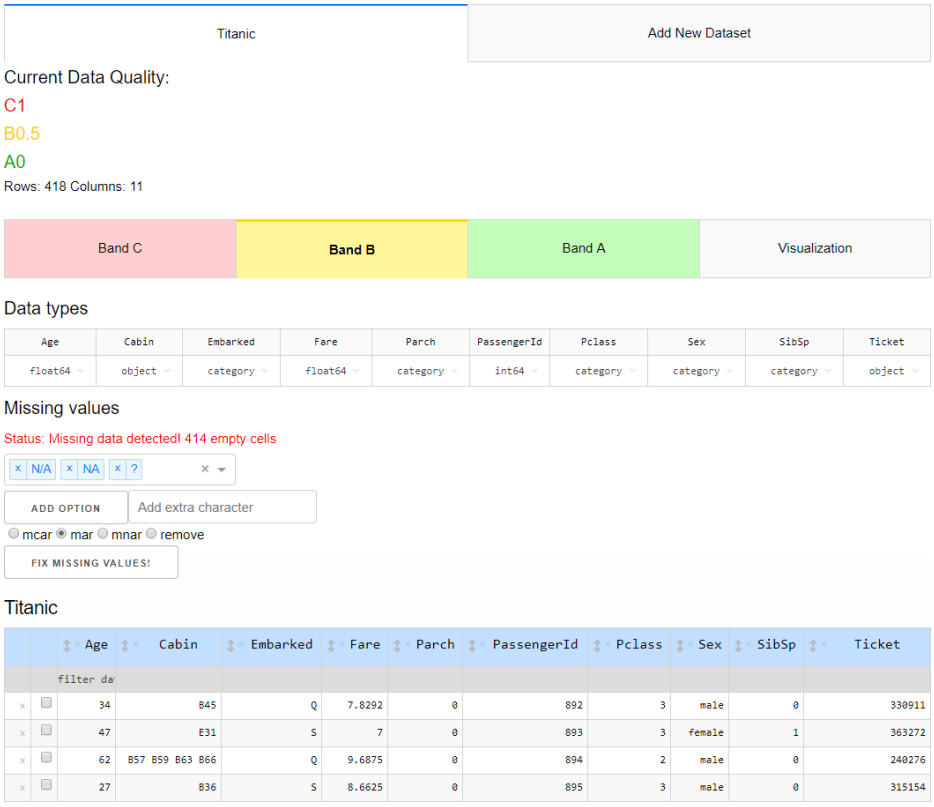
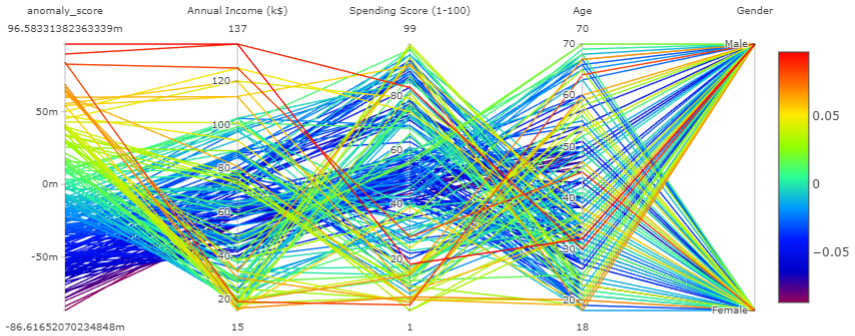


Fig. 1. The PyWash web interface.

In addition, we built a user interface on top of the python package in the form of a web application to guide the user through the cleaning process. The interface, shown in Figure 1, groups the operations for each band into separate color-coded tabs. Loading datasets happens in the tab called 'Band C', which will initially be the only band that is accessible. Band C includes interactions (e.g. forms) to import datasets and merged them together if applicable. Once a dataset is successfully parsed, it will receive a tab at the top of the screen with its name and the other bands will be unlocked. The interface supports multiple datasets being loaded at the same time, with every loaded dataset in a unique tab. When a dataset is selected, a description is shown which includes the data quality score. Below this description, there is a row of buttons which are used to switch between the bands from the framework. Selecting different bands will show the various operations available to fix or investigate the deficiencies that are part of that band. Band B is selected in Figure 1. Data types are automatically inferred and shown to the user, who can leave them as is, or correct them through a dropdown menu. In the missing value section, an indicator shows whether missing values are detected. Users can also check the data in the shown table



Mall_Customers

		Age	Annual Income (k$)	Gender	Spending Score (1-100)	anomaly_score	prediction
		filter di					
×	☐	32	137	Male	18	0.09658331382363339	1
×	☐	30	137	Male	83	0.08973938934859016	0
×	☐	64	19	Male	3	0.08401876579198053	0
×	☐	45	126	Female	28	0.08274928973733486	0
×	☐	20	16	Female	6	0.06875584617998964	0

Fig. 2. Visualization of outliers in PyWash, color-coded by anomaly score.

add extra characters that indicate a missing value. Missing values can optionally be imputed or removed. At the moment, the user still has to choose between one of four techniques (based on whether the data is missing at random or not), but we hope to automate this further in future work.

Underneath the operations from the selected band, a data table is shown so that the effect of the operations is immediately visible. This table supports row/column deletion, filtering, sorting, and editing. The exporting options will store return the data 'as is', thus including any modifications made in the table (filtering, sorting, etc.). In future work, we also plan to export a log of all cleaning operations with the data export.

The rightmost tab contains a visualization section to help the user with decision making and perform instant exploratory analysis. Figure 2 is an example of a parallel coordinates graph color-coded by an anomaly score computed by an Isolation Forest in band A. The outlier detection adds the columns 'anomaly_score' to the plot and a 'prediction' column to the table indicating which rows are predicted to be outliers. The first row is marked as an outlier and highlighted. Since these columns are added as part of the dataset, we can export them for further analysis with other tools and visualizations.

Our data quality framework, the PyWash package, and the web interface are all open source and we warmly welcome and encourage anyone to help fine-tune the framework and improve the libraries and interfaces.

5 Discussion

The goal of this paper is to streamline the data cleaning process by creating a vocabulary and a framework for automatic data cleaning, such that the data cleaning process becomes an explicit, accountable, reported process. This will help in communicating and describing the quality of your data to others and acting on it adequately.

Although there are many challenges still to be resolved, we hope that this work contributes to more standardized and automated cleaning processes. Most of the deficiencies in band C can already be automated to a large extend. Automated decoding, parsing and storing of datasets can be done reliably, while data formats and disjoint datasets can be detected when enough data is available. Band B can also be automated to some degree since data types, missing values, and duplicated records can be detected and issues can be (partially) resolved, as long as the user specifies how to solve it.

Unfortunately, not all deficiencies can be automatically detected and fixed. In some cases, domain expertise and common sense reasoning are essential. Bias detection tools do exist, but most band AA deficiencies will have to be checked by humans since detecting the context of a dataset is often impossible or unreliable. Therefore, we have taken a human-in-the-loop approach that provides as much guidance and automation as possible, yet leaves many decisions at the discretion of the user.

From a usage perspective, an open challenge is that we have no objective method to determine the weights and thresholds of each band. We have supplied default values, but these will not suffice for everyone since not every dataset and analysis has to meet the same requirements. Also, the framework does not yet encompass every aspect of data preprocessing. For instance, feature construction would be a valuable addition to band A, but much more work is needed to codify it and guide the user in applying it.

However, we do already provide an extensible software framework in which new techniques that automate data cleaning and preprocessing can be implemented and made easily available to anyone. As such, we hope that it will become a test bed for automated data science research in general.

Acknowledgements. The authors would like to thank Neil D. Lawrence for contributing many original ideas and his valuable feedback on the ideas in this paper. We also like to thank Hero de Smeth for contributing to the definition and composition of the bands.

References

1. Allison, P.D.: Missing Data. SAGE, Thousand Oaks [Calif.]; London (2002).
2. Arslan, R.C.: How to Automatically Document Data With the codebook Package to Facilitate Data Reuse. Advances in Methods and Practices in Psychological Science. 2, 169187 (2019). https://doi.org/10.1177/2515245919838783.

3. Castelijns, L.A., Maas, Y.: PyWash. (2019). https://github.com/pywash/pywash
4. Dandurand, L., Serrano, O.S.: Towards improved cyber security information sharing. In: 2013 5th International Conference on Cyber Conflict (CYCON 2013). pp. 116 (2013).
5. General Data Protection Regulation, European Union, Regulation (EU) 2016/679 (2018). https://gdpr-info.eu/
6. Guyon, I., Elisseeff, A.: An Introduction to Variable and Feature Selection. J. Mach. Learn. Res. 3, 11571182 (2003).
7. Kenett, R., Shmueli, G.: Information quality: the potential of data and analytics to generate knowledge. Wiley, Chichester, West Sussex (2017).
8. Khosla, A., Zhou, T., Malisiewicz, T., Efros, A.A., Torralba, A.: Undoing the Damage of Dataset Bias. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., and Schmid, C. (eds.) Computer Vision ECCV 2012. pp. 158171. Springer Berlin Heidelberg, Berlin, Heidelberg (2012). https://doi.org/10.1007/978-3-642-33718-5_12.
9. Kim, W., Choi, B.-J., Hong, E.-K., Kim, S.-K., Lee, D.: A Taxonomy of Dirty Data. Data Mining and Knowledge Discovery. 7, 8199 (2003). https://doi.org/10.1023/A:1021564703268.
10. Kriegel, H.-P., Krger, P., Zimek, A.: Outlier Detection Techniques. Tutorial, 16th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Washington DC (2010).
11. Krishnan, S., Haas, D., Franklin, M.J., Wu, E.: Towards Reliable Interactive Data Cleaning: A User Survey and Recommendations, (2016). https://sirrice.github.io/files/papers/cleaning-hilda16.pdf
12. Krishnan, S., Franklin, M.J., Goldberg, K., Wang, J., Wu, E.: ActiveClean: An Interactive Data Cleaning Framework For Modern Machine Learning. In: Proceedings of the 2016 International Conference on Management of Data - SIGMOD 16. pp. 21172120. ACM Press, San Francisco, California, USA (2016). https://doi.org/10.1145/2882903.2899409.
13. Lawrence, N.D.: Data Readiness Levels. arXiv:1705.02245 [cs]. (2017).
14. Li, S., Momoi, K.: A composite approach to language/encoding detection. In: 19th International Unicode Conference (2001).
15. Patterson, J.B.: Coded Character Sets, History and Development. IEE Proc. E Comput. Digit. Tech. UK. 128, 173 (1981). https://doi.org/10.1049/ip-e.1981.0034.
16. Reis, M.S., Kenett, R.: Assessing the value of information of data-centric activities in the chemical processing industry 4.0. AIChE J. 64, 38683881 (2018). https://doi.org/10.1002/aic.16203.
17. Valera, I., Ghahramani, Z.: Automatic Discovery of the Statistical Types of Variables in a Dataset. In: Proceedings of the 34th International Conference on Machine Learning - Volume 70. pp. 35213529. JMLR.org (2017).
18. Yergeau, F.: UTF-8, a transformation format of ISO 10646. (2003).