Contextualisation of Datasets for better classification models: Application to Airbus Helicopters Flight Data

Marie Le Guilly, Nassia Daouayry, Pierre-Loic Maisonneuve, Ammar Mechouche, Jean-Marc Petit, Vasile-Marian Scuturici

Contextualisation of Datasets for better classification models: Application to Airbus Helicopters Flight Data

Marie Le Guilly[1], Nassia Daouayry[1,2], Pierre-Loic Maisonneuve[2], Ammar Mechouche[2], Jean-Marc Petit[1], and Vasile-Marian Scuturici[1]

[1] Univ Lyon, INSA Lyon, LIRIS (UMR 5205 CNRS), Villeurbanne, France
{marie.le-guilly, jean-marc.petit, marian.scuturici}@insa-lyon.fr
[2] Airbus Helicopters, Marignane, France
{nassia.daouayry, pierre-loic.maisonneuve, ammar.mechouche}@airbus.com

Abstract. For helicopters, anticipating failures is crucial. To this end, the analysis of flight data allows to develop predictive maintenance approaches, for which Airbus Helicopters (AH) has proposed several solutions, some based on machine learning using predictive models. One recurrent problem in this setting is the *contextualization* of the data, that is to identify the data better fitting the phenomenon being modeled. Indeed, helicopters are complex systems going through different flight phases. Experts therefore have to identify the adequate ones, in which the selected flight parameters are stable and consistent with the studied problem. In this paper, we propose a generic solution to contextualize classification data, and present an experimental study on AH flight data: the results are encouraging and allow to keep domain experts involved the process.

Keywords: data contextualization, failure anticipation, classification

1 Introduction

In the helicopters industry, predictive maintenance is crucial and Airbus Helicopters (AH) seeks to anticipate failure as soon as possible. One solution is to analyze flight data, as most helicopters are equipped with flight recorders for hundreds of parameters. Such an amount of data makes it possible to analyse "low-level signals"' over longer periods of time, and to detect failures earlier. In this context,AH has gathered data on hundreds of thousands flight hours: to face such a huge amount of data, a Big Data platform has been deployed at AH to enable the storing and processing of large quantities of data [9].

Using this platform, *digital twins* have been devised to identify as soon as possible small variations on core physical sensors. They are mainly based on physical models and expert knowledge, but AH combines these with machine learning techniques to build predictive models from the data. To build such models, AH faces generic and recurrent issues that are well-known in machine

learning, such as data cleaning, accuracy , or explainability. But in addition to these classic issues, AH also seeks to build models corresponding to the *normal behavior* of the system, and has to use data fitting the behavior algorithms have to model. Indeed, an important filtering step is performed to identify the data that is adequate to deal with the considered problem: complex systems such as helicopters go through many different phases, and only a subset of the data is relevant for a given model, as they are the only one for which the laws of normal behavior of the system apply. It is therefore necessary to identify the correct context for the considered task, which is the subset of data corresponding to the desired phases on which the model is applied. We define this problem as *contextualization*, according to the term used by AH experts. Thus, it consists in determining the flight phases where considered parameters have lesser variability and are less subject to pilot maneuvers and external parameters not recorded by the system. At AH, this crucial step is dealt with by relying on experts knowledge who specify how to filter flight data.

The contextualization problem can seem as a simple problem at first hand (mainly data selection), but turns out to be a nightmare in practice. Identifying the appropriate data is clearly not an easy task, and depends on the final objective for the classification model. In addition, contexts are tightly linked with the application they concern, so solutions are often specific to one given situation. For systems such as helicopters, contextualization is also important as they are systems governed by physical laws, that apply only in specific contexts: the purpose of classification models is therefore to produce outputs coherent with these laws. To this end, these models have to be trained on data consistent with the physical model they represent.

In this paper, we propose our ongoing work to address the contextualization challenge. We seek to identify the appropriate context for a classification task, by identifying the subset of data more likely to capture the normal behavior. To do so, we seek the data favoring the existence of a function between the features and the class to predict. As the correct context should follow some underlying function the model seeks to define, we propose to remove the regions of the data preventing the existence of that function, and to only keep the data more likely to correspond to a normal behavior. We then show how this approach can be applied to AH classification datasets.

Based on these considerations, we made the following contributions: (1) Proposing a generic solution for contextualization, in order to define filters that can be used to reduce the dataset to a given context; (2) Experiments on AH data showing how identifying context elements can improve the accuracy of classifiers; (3) Confronting a contextualization proposed by AH experts to additional context elements proposed by our method.

Section 2 introduces the preliminaries. In section 3, we propose our approach to better contextualize datasets, and in section 4, we focus on AH's data, to show how we built a context for the considered dataset, and develop the lessons drawn from this collaboration based on the experimentations that have been conducted. Finally section 5 presents the related work before concluding in section 6.

2 Preliminaries

2.1 Functional dependencies

We first recall basic notations and definitions (see [8]). Let U be a set of attributes. A relation schema R is a name associated with attributes of U, i.e. $R \subseteq U$. A database schema $\mathcal{R}$ is a set of relation schemas. Let D be a set of constants, $A \in U$ and R a relation schema. The domain of A is denoted by $dom(A) \subseteq D$. A tuple t over R is a function from R to D. A relation r over R is a set of tuples over R. If $X \subseteq U$, and if t is a tuple over U, then we denote the restriction of t to X by $t[X]$. If r is a relation over U, then $r[X] = \{t[X], t \in R\}$.

Definition 1. *Let R be a relation schema, $X \subseteq R$ and $C \subseteq R \backslash X$. A FD on R is an expression of the form $R : X \rightarrow C$ (or simply $X \rightarrow C$ when R is clear from context)*

Definition 2. *Let r be a relation over R and $X \rightarrow C$ a functional dependency on R. $X \rightarrow C$ is satisfied in r, denoted by $r \models X \rightarrow C$, if and only if for all $t_1, t_2 \in r$, if $t_1[X] = t_2[X]$ then $t_1[C] = t_2[C]$.*

2.2 Supervised classification in machine learning

Let's consider a set of N training samples $\{(x_1, y_1), ..., (x_N,\ y_N)\}$ where x_i is the feature vector of the i-th example and y_i its label (or class). The number of different labels K, is limited and much smaller than the number of samples.Given this, classification is the task of learning a target function g (a classifier) that maps each example x to one of the k classes, with the lowest error rate. It is possible to express a classification problem using relational databases notations. In the sequel, we will therefore consider a relation $r_0(A_1, \ldots, A_n, C)$ with N tuples, where for any tuple t_i, $t_i[A_1 \ldots A_n] = x_i$ and $t_i[C] = y_i$. In addition, we consider that traditional feature selection methods (see [1]) have been applied and consider the subset $X \subseteq \{A_1 \ldots A_n\}$ of selected features.

To evaluate the performances of an algorithm, we use accuracy, which is the proportion of samples that are correctly classified by a model. This score lies between 0 and 1, and ideally should get as close as possible to 1. Given a model M over a relation r, accuracy is defined as follows:

$$accuracy(M, r) = \frac{\#\, of\, correct\, predictions}{|r|}$$

2.3 Existence versus determination of a function

We use the link between FDs and classification, developed in [7]. We only underline here it relies on the notion of function, as classifier seeks to define a function from the features to the class, while the FD $X \rightarrow C$ can say whether or not such a function exists or not: the FD $X \rightarrow C$ is satisfied if and only if there exists a function from X to C. If the FD is not satisfied, it means some pairs of tuples have the same value on X, but different classes. Such tuples are called counterexamples:

Definition 3. *Let r be a relation over R and $X \rightarrow C$ a FD f on R. The set of counterexamples of f over r is denoted by $CE(X \rightarrow C)$ and defined as follows:*

$$CE(X \rightarrow C, r) = \{(t_1, t_2) | t_1, t_2 \in r, t_1[X] = t_2[X] \text{ and } t_1[C] \neq t_2[C]\}$$

Counterexamples are important as they identify pairs of tuples for which the classifier cannot perform correctly, as for the same input, it always predicts the same output. The proportion of counterexamples therefore directly impacts the quality of the classification: it can be evaluated using measure G_3, and contrary to [5] that presents this measure as an error, we propose it as follows:

$$G_3(X \rightarrow C, r) = \frac{max(\{|s| | s \subseteq r, s \models X \rightarrow C\})}{|r|}$$

Measure G_3 is of crucial importance for the classification problem, as in the subset s defined for G_3, there exists a function between the left and right hand side of the dependency. For classification, measure G_3 is therefore a way to bound the accuracy a classifier can reach on the considered dataset, as it is necessary limited by the existence of counterexamples. As a result, the following result holds, for which the details and proof are given in [7]:

Proposition 1. *Let $X \subseteq R$ be a set of features, $C \in R$ the class to be predicted, r a relation over R, and M a classifier from X to C. Then:*

$$accuracy(M, r) \leq G_3(X \rightarrow C, r)$$

In the setting of contextualization, G_3 can be seen as a way to identify whether or not a dataset follows a function, and to identify zones that are therefore more likely to correspond to a normal behavior of the system.

3 Contextualization of a classification dataset

The objective is to propose a methodology for the contextualization of classification datasets. The proposed solution considers there should be a function between the features and the class to predict. The idea is to identify the regions in the initial dataset in which a function is likely to exist, and therefore in which the FD $features \rightarrow class$ is likely to be satisfied. On the opposite, regions with a high proportion of counterexamples should be removed, as they are likely regions where the model hypothesis are not verified.

To contextualize a dataset, we propose an iterative approach, that is summarized on figure 1. The process starts with an initial classification dataset. It is then discretized, to smooth the data variability and to better identify counterexamples. Then, G_3 is computed, and a classifier is trained and tested, to obtain an accuracy measure. Measure G_3 allows to evaluate the existence of a function, while the accuracy guaranties the performances of the model. These two measures are taken into account to determine the next step in the process.

If the domain experts are not satisfied with the measures, the counterexamples are enumerated, to identify filters to remove the tuples that cause too many counterexamples. The key is to find balance between removing regions of the data while keeping as many tuples as possible. The filters can take different forms: here, we propose to define filters in the form of conjunctions of conditions allowing to remove groups of tuples. To identify such groups, visualizations are proposed, to observe what tuples are the most involved in counterexamples. Once the filters are determined, based on these visualizations and in collaboration with domain experts, tuples are removed, providing a new dataset. This process is repeated until satisfaction.

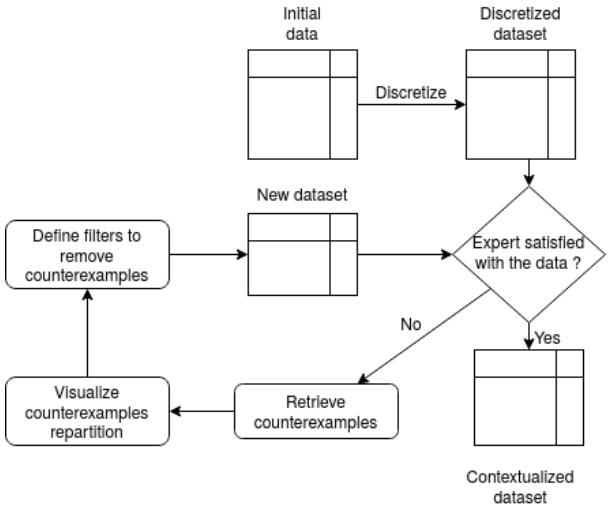


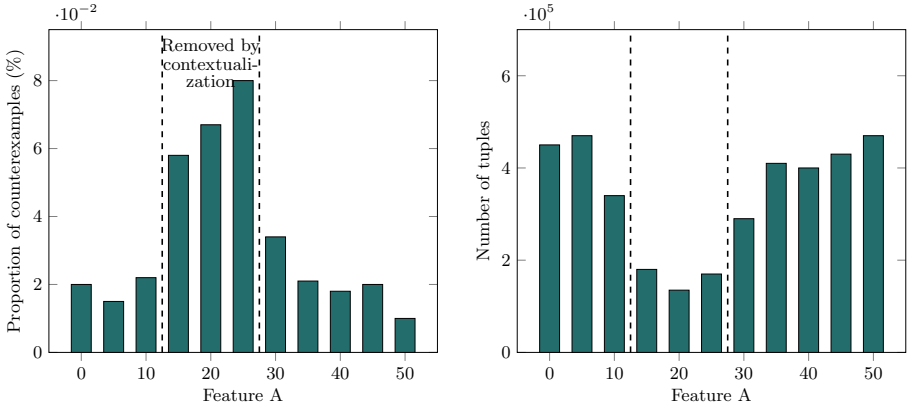
Fig. 1: Overview of the solution proposed to contextualize a classification dataset

3.1 From counterexamples to context-aware data selection

When the proposed contextualization is not satisfying, solutions have to be proposed to refine it, and to therefore remove tuples from the dataset. The challenge is to determine what are the tuples to remove and why. We therefore propose to determine filters that can be applied to the dataset, to remove tuples and lower the number of counterexamples in the dataset. Such filter should ideally remove as few tuples as possible, while removing as many counterexamples as possible. Indeed, one tuple might be involved in many counterexamples: in this case, it should be removed.

Many solutions can be considered for the filters: one solution from example is too order the tuples by the number of counterexamples they are involved in, and to set a threshold to remove all the tuples involved in more counterexamples than

this threshold. But it does not explain what are the characteristics of the removed tuples: if a domain expert wishes to understand why a tuples is removed, she has to manually check each counterexample. In this paper, we propose to define filters in the form of conjunction of conditions applied to the dataset, making the overall process explainable. These filters define, in simple terms, regions of the dataset containing more counterexamples than others, while concerning only a few tuples. This can be performed using visualizations proposing, for each feature, histograms showing the distribution of values among counterexamples, and the number of tuples taking a given value. The histograms can then be used to identify values having, on a given feature, few tuples involved in many counterexamples. The filters then integrate a condition removing such values from the dataset. Such filters are interpretable by domain experts, who can analyze whether or not these filters make sense with the desired context.



(a) Counterexamples proportion for each value of a given feature

(b) Histogram of values frequency for a given feature

Fig. 2: Toy example for filter design

Example 1. Figure 2 presents visualizations used to define filters. For a given feature A, figure 2a shows for each value taken by this feature, the proportion of tuples involved in counterexamples, and therefore how much they contribute to the value of G_3. Figure 2b is an histogram of values for the considered feature. By comparing these two visualizations, it appears there is a zone that does not contain many tuples, but many counterexamples. As a result, one condition for a contextualization filter could be to remove all tuples for which $A \geq 15$ and $A \leq 25$. This gives an interpretable filter, removing a few tuples and improving measure G_3. Similar work can be performed for each feature of the dataset, creating a filter that is a conjunction of conditions over all features.

4 Application to AH flight data

4.1 AH classification datasets

Using helicopters flight data, AH is developing tools such as virtual sensors, that aim at monitoring the aircraft health and usage. They use the historical flight data to learn a predictive model for a given parameter. The predicted value is compared to the one given by the physical sensor: an alert is raised if the difference between the two values is too high. An example of such a virtual sensor has been proposed by AH for the oil pressure of the helicopter Main Gear Box (MGB) [2]. We reuse the data from this study to perform the experiments of this paper. As a first contextualization had been done by AH domain experts, we used and compared two datasets, with 10 attributes selected and discretized by AH experts: the **raw dataset** corresponds to the flight data without any contextualisation, for a given period of time, randomly mixing tuples from several flights; the **expert-Contextualized dataset** is a subset of the raw one containing tuples filtered by AH experts (around 50% of the raw data).

4.2 Comparison of AH datasets

	Baseline		Filter 1		Filter 2	
Dataset	# tuples	accuracy	# tuples	accuracy	# tuples	accuracy
Raw	1969533	53.97%	607248	57.28%	468630	61.71%
Expert-contextualized	541342	73.94%	281947	76.02%	100165	78.61%

Table 1: Accuracy of random forest models on the oil pressure datasets

The impact of contextualization was analyzed, by comparing accuracy for a random forest algorithm (*baseline* column of table 1). The accuracy for the expert-contextualized dataset is much higher than for the raw one,confirming the expert contextualization pertinence. Moreover, $G_3 = 95.53\%$ for raw dataset and $G_3 = 95.51\%$ for expert-contextualized one. The proportion of counterexamples is therefore reasonable and the two datasets have similar G_3 values. The contextualization seems to have preserved the proportion of counterexamples: they have decreased in absolute number, but not with respect to the size of the dataset. New contextualization might therefore increase the model's accuracy.

4.3 Additional contextualization using G_3

We applied our methodology from figure 1 to the two datasets, but first verified that the counterexamples were evenly distributed among the flights. Figure 3 shows a histogram of the percentage of counterexamples among flights: most flights have a very low rate of counterexamples, so any removal of counterexamples affects a large number of flights, avoiding the model to overfit on a subpart

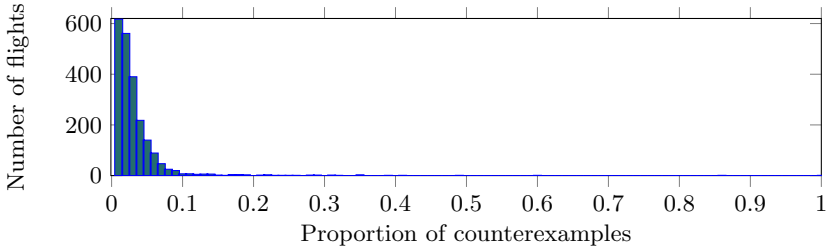


Fig. 3: Distribution of flights for each proportion of counterexamples

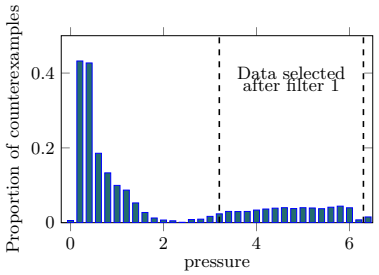
of the flights. We then analyzed the two plots made for each feature such as figures 4a and 4b for the pressure. Low pressure values have more counterexamples, while containing an important number of tuples. It can also be noted that the domain contextualization removes a significant part of counterexamples, but other regions could be cleaned further from counterexamples with additional contextualization, for example for pressure values over 5.6 bar.

A first filter was designed (Filter 1 in table 1). For the pressure, this filter removes all the data for which it is below 3.2 bar and above 6.4 bar, as these regions have few tuples but many counterexample (see figure 4). Similar rules were applied for the other features of the dataset. The results in table 1 show the positive effect of this filter on classifier's accuracy. It was decided to improve again the contextualization, so we obtained filter 2 by adding additional rules to the ones from filter 1. Table 1 shows that accuracy is improved by filter 2. After this second iteration, the obtained contextualization was considered satisfying.

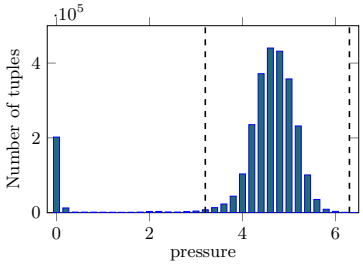
Finally, it should be noted from table 1 that there is a significant gap between the highest accuracy on the raw dataset and the lowest accuracy for the expert-contextualized one. Even with the best filter, it is not possible to reach the result obtained using expert knowledge: the best approach consists in taking the valuable domain expert knowledge into account, before refining it using tools such as counterexamples and G_3.

4.4 Take away lessons

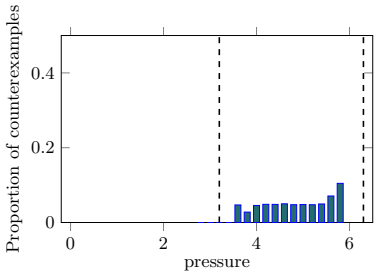
These experimentation showed how contextualization can be used to improve the accuracy of classifiers for AH virtual sensors. Contextualization is an important problem, but it is not easy to address because the proposed solutions are often domain-specific, or included in the "data preparation" steps that our left to data scientists judgment: our solution could in comparison be applied for other types of application and involves domain experts in the loop. There is also a qualitative aspect to this approach, that aims at taking a step back from the model, to understand what is being done, and understand the limitations. This is directly related to the explicability of the model, a crucial notion in aeronautics: the prediction of what can be seen as a simple classification algorithm output can put into question human lives getting back into an aircraft or not.



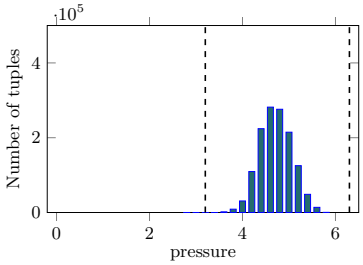
(a) Raw: counterexamples proportion against pressure



(b) Raw: histogram of pressure values



(c) Expert-contextualized: counterexamples proportion against pressure



(d) Expert-contextualized: histogram of pressure values

Fig. 4: Counterexamples and distribution for pressure values

5 Related work

We applied our contextualization technique in the context of predictive maintenance for helicopters, a growing topic in the industry. Virtual sensors such as the ones used for the experiments of this paper [2] are interesting solutions in this context. Similarly, [3] proposes a virtual sensor to anticipate failures on photo-voltaic systems. Additionally, [11] presents a failure anticipation approach for aircraft systems. In this case, the learning is done only on flight phases predefined by experts. More generally, in most works developed in the industry, data is always combined with domain knowledge in order to speed-up accurate predictive models development. However, this combination still is often not optimal, and we believe this is a lever for improving accuracy of predictive models developed in the industry.

Functional dependencies are of high interest for data cleaning, a necessary prerequisite for data contextualization. The authors from [6] showed that if there is a functional dependency between features, it is likely to affect the classifier negatively. Specific dependencies have been proposed to identify inconsistencies

in a dataset, and eventually repair it. Matching dependencies [4] for data repairing uses matching rules to relax the equality on functional dependencies and assign values for data repairing. In Holoclean [10], dependencies are used to clean automatically a dataset.

6 Conclusion

In this paper, we addressed the problem of contextualization of classification datasets, applied to the flight data of AH. This problem is crucial, and appears in many data science industrial applications, but has yet not been addressed as massively as other traditional machine learning problems. We proposed a methodology, and conducted experiments on data from a virtual sensor developed by AH, and showed how our method could improve the contextualization and, as a consequence, the accuracy of the datasets.

References

1. Arauzo-Azofra, A., Aznarte, J.L., Benítez, J.M.: Empirical study of feature selection methods based on individual feature evaluation for classification problems. Expert Systems with Applications **38**(7), 8170–8177 (2011)
2. Daouayry, N., Mechouche, A., Maisonneuve, P.L., Petit, J.M., Scuturici, M.: Data-centric helicopter failure anticipation: The mgb oil pressure virtual sensor case. In: International Conference on Big Data. p. 10 pages. IEEE (2019)
3. De Benedetti, M., Leonardi, F., Messina, F., Santoro, C., Vasilakos, A.: Anomaly detection and predictive maintenance for photovoltaic systems. Neurocomputing **310**, 59–68 (2018)
4. Fan, W.: Dependencies revisited for improving data quality. In: Proceedings of the twenty-seventh ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems. pp. 159–170. ACM (2008)
5. Kivinen, J., Mannila, H.: Approximate inference of functional dependencies from relations. Theoretical Computer Science **149**(1), 129–149 (1995)
6. Kwon, O., Sim, J.M.: Effects of data set features on the performances of classification algorithms. Expert Systems with Applications **40**(5), 1847–1857 (Apr 2013). https://doi.org/10.1016/j.eswa.2012.09.017
7. Le Guilly, M., Petit, J.M., Scuturici, V.M.: Evaluating classification feasability over datasets using functional dependencies. In: BDA 2019 35ème conférence sur la Gestion de Données: Principes, Technologies et Applications. Lyon, France (2019)
8. Levene, M., Loizou, G.: A guided tour of relational databases and beyond. Springer Science & Business Media (2012)
9. Mechouche, A., Daouayry, N., Camerini, V.: Helicopter big data processing and predictive analytics: Feedback and perspectives. In: Proceedings of the 45th European Rotorcraft Forum. p. 7 pages. Warsaw, Poland (2019)
10. Rekatsinas, T., Chu, X., Ilyas, I.F., Ré, C.: Holoclean: Holistic data repairs with probabilistic inference. Proceedings of the VLDB Endowment **10**(11), 1190–1201 (2017)
11. Sundareswara, R., Betz, F.D., Lu, T.C.: Interpretable unsupervised feature extraction and learning of abnormal system state transitions in aircraft sensor data. In: Proceedings of the Annual Conference of the PHM Society. vol. 10 (2018)