Romano, Stefania; Martinez-Heras, Jose; Natalini Raponi, Francesco; Guidi, Gregorio; Gottron, Thomas

Working Paper

Discovering new plausibility checks for supervisory data: A machine learning approach

ECB Statistics Paper, No. 41

Provided in Cooperation with:
European Central Bank (ECB)

Suggested Citation: Romano, Stefania; Martinez-Heras, Jose; Natalini Raponi, Francesco; Guidi, Gregorio; Gottron, Thomas (2021) : Discovering new plausibility checks for supervisory data: A machine learning approach, ECB Statistics Paper, No. 41, ISBN 978-92-899-4700-8, European Central Bank (ECB), Frankfurt a. M., https://doi.org/10.2866/19338

This Version is available at:
https://hdl.handle.net/10419/274089

EUROPEAN CENTRAL BANK

EUROSYSTEM

Statistics Paper Series

Stefania Romano, Jose Martinez-Heras,
Francesco Natalini Raponi, Gregorio Guidi,
Thomas Gottron

Discovering new plausibility checks for supervisory data

A machine learning approach

No 41 / October 2021

Contents

Abstract

In carrying out its banking supervision tasks as part of the Single Supervisory Mechanism (SSM), the European Central Bank (ECB) collects and disseminates data on significant and less significant institutions. To ensure harmonised supervisory reporting standards, the data are represented through the European Banking Authority's data point model, which defines all the relevant business concepts and the validation rules. For the purpose of data quality assurance and assessment, ECB experts may implement additional plausibility checks on the data. The ECB is constantly seeking ways to improve these plausibility checks in order to detect suspicious or erroneous values and to provide high-quality data for the SSM.

In this paper we describe a data-driven approach, based on machine learning, for discovering new plausibility checks. Specifically, the approach makes use of large amounts of historical data to identify patterns in past observations. The patterns of interest correspond to latent and potentially non-linear relationships in the data, which serve as a basis for defining new checks. We show that this approach can be used to detect relevant patterns and that these patterns are suitable for discovering anomalies in the data. We also illustrate how such patterns are used by business experts to refine their data quality framework. We finally provide suggestions for potential further work that could be carried out to improve technical performance as well as prediction quality

JEL codes: C18, C63, C81, E58, G28

Keywords: machine learning, quality assurance, validation rules, plausibility checks, supervisory data

1 Introduction

In carrying out its banking supervision tasks as part of the Single Supervisory Mechanism (SSM), the European Central Bank (ECB) makes use of data provided by national competent authorities (NCAs) to assess the health of financial institutions within the euro area. Credit institutions report their data in line with the reporting requirements defined by the European Banking Authority (EBA) through its data point model (DPM)[1]. Within the DPM, the EBA defines validation rules to ensure the correctness, completeness and consistency of data. The ECB has its own system – the Supervisory Banking data system (SUBA) – for collecting data. SUBA allows supervisory data to be collected from the NCAs and implements all of the EBA's validation rules to assess the data quality. Furthermore, ECB experts can carry out additional plausibility checks to ensure the quality of the data. To date, such plausibility checks have been defined according to a knowledge-driven approach, with business experts using domain-specific knowledge and insights into business processes at supervised institutions.

A major drawback of relying on experts' knowledge alone is that it is impossible for a human expert to assess all of the possible ways in which data points – the variables in this analysis – might be related. There are around three million variables that may potentially be reported for every institution and reference period. The possible combinations of relationships between two or more variables that might need to be investigated could therefore run into the millions. In this paper, we describe a complementary approach for discovering new plausibility checks. The novelty of the approach is that it is data-driven. This means it makes use of large amounts of historical data to identify patterns in past observations. The patterns of interest correspond to latent and potentially non-linear relationships in the data.

To identify latent patterns in the data, we train multiple regression models – one model for each observed data point. The approach is capable to incorporate domain knowledge to avoid identifying trivial and already known relationships, e.g. relationships defined in existing validation rules. The regression model for a specific data point uses other observed values to make a prediction of what value to expect. The prediction, which is flexible and error-tolerant, indicates an interval that is expected to contain the value. By design, the approach also provides insights into which other observed data points are among the main contributing factors and explain the expected values. In this way, the models not only provide a "black box" prediction but allow an analysis of the underlying relationships discovered between the observed data points. These insights can be used by business experts to formulate new plausibility checks. In addition, the approach lends itself to providing a normalised implausibility score for each observation, measuring the degree to which an observation deviates from its expected value. This normalised score makes it possible to compare observations for different data points and incorporates knowledge about the natural variance and noise in the data. By aggregating the

[1] See the EBA's website for information on the DPM data dictionary.

normalised degree of deviation across all values of an institution (or “entity”), we are also able to provide a consolidated outlier score for the entire report of an entity, giving data quality managers a tool for assigning priorities in their assessment of data received.

In this paper, we motivate our approach and describe it in detail. We illustrate the necessary steps in data collection and pre-processing. We describe how to incorporate prior domain knowledge into the models, which is essential for the detection of non-trivial relationships. Finally, we present the methods for calculating the implausibility score at the data point and entity levels and show how such a score can be used by data quality managers to assign priorities in their work. In a qualitative evaluation, we demonstrate the effectiveness of the approach and provide initial evidence on how the approach can be used by quality managers to gain deeper insights – and eventually also design new business-motivated quality checks.

The rest of the paper is structured as follows. In Section 2, we provide detailed background information on the collection of supervisory data at the ECB and on the quality mechanisms that are already in place. We then go on, in Section 3, to look at related work relevant to our approach. In Section 4, we describe the idea of modelling the discovery of plausibility checks as a machine learning task, providing the formal foundation for the implementation we discuss in Section 5. We address the experimental set-up and evaluation of our approach in Section 6, before concluding the paper with a summary and a look ahead to future work.

2 Background

After the 2008 financial crisis, the heads of state and government of euro area countries decided to create an EU banking union to enhance the resilience of banks. In particular, they announced the creation of the Single Supervisory Mechanism (SSM), a framework within which ECB Banking Supervision (the supervisory arm of the ECB) – in collaboration with the national competent authorities (NCAs) – plays a supervisory role in monitoring the financial stability of the banks in the European Union (Detken and Nymand-Andersen, 2013). Through this mechanism, the ECB is responsible for the supervision of euro area banks, as well as banks of EU countries outside the euro area that have voluntarily decided to participate in the SSM.[2] The ECB works closely with the NCAs, performing tasks that differ depending on the role and significance of the supervised bank. Banks are divided into significant institutions (SIs) and less significant institutions (LSIs). This distinction is mainly based on size, economic importance and scope of cross-border activities. SIs are supervised directly by the ECB through Joint Supervisory Teams (JSTs) comprising staff of the ECB and of the NCAs. Supervision of LSIs is delegated to the NCAs in accordance with the principle of proportionality.

The European Commission has instructed the European Banking Authority (EBA) to define Implementing Technical Standards (ITS)[3] for the supervisory reporting requirements. The ITS on Supervisory Reporting provide a consistent, repeatable, standardised method for information sharing. The ECB collects these data through the Supervisory Banking data system (SUBA) to support consistent supervision within the euro area. SUBA provides all relevant information for supervisory purposes in a central platform. Subsets of the data are disseminated to other systems, depending on the business needs of the counterparties. For instance, JSTs supervising SIs only have access to SI data. Therefore, SUBA allows a complete overview of supervisory data and represents a good basis for the approach presented in this paper.

2.1 Collection of supervisory data: purpose and process

Supervisory data are submitted by each credit institution to the competent NCA. The NCA in turn sends these values to the ECB through the SUBA system. The SUBA platform allows data to be collected, processed, aggregated and disseminated to several counterparties. Subsets of the data are then disseminated both within the ECB and to other European institutions such as the EBA and Single Resolution Board. This chain of reporting is referred to as the “sequential approach”. SUBA represents the point of the reporting chain where all the information comes together and is integrated in one place. SUBA thus provides a comprehensive view of the

2 Two non-euro EU countries, Bulgaria and Croatia, joined the SSM at the end of 2020. This study was conducted before these countries joined the SSM, therefore data of institutions resident in those countries is not in the scope of this paper.

3 Implementing Technical Standards on Supervisory Reporting.

data and is the most suitable place in the reporting chain for data-driven analytics and machine learning.

SUBA data are of interest for several business areas within the ECB. In particular, ECB Banking Supervision bases its evaluation of banks' financial health on the data relating to the SIs and LSIs. Therefore, given the criticality of the SUBA data use cases, data quality is of utmost importance in SUBA.

All the data are collected in templates (grouped into modules) in accordance with the technical standards defined by the EBA in order to implement uniform reporting requirements; this makes data comparable, allowing for more efficient supervisory activity. The EBA technical standards (formalised by EU Regulation No 680/2014[4]) reflect the reporting obligations embedded in the Capital Requirements Regulation (EU Regulation No 575/2013[5]) and cover reporting of own funds and capital requirements, reporting of financial information, reporting on large exposures, reporting on leverage and reporting on liquidity and stable funding. They are complemented by other specific reporting templates such as asset encumbrance, forbearance and non-performing exposures. Reporting is carried out under the common reporting (COREP) and financial reporting (FINREP) frameworks. These were developed by the Committee of European Banking Supervisors (the predecessor of the EBA) and cover the following information.

- COREP is the framework for reporting basic regulatory information. It covers six reporting areas for capital adequacy and capital requirements: capital adequacy, group solvency/large exposures, credit risk, operational risk, market risk and liquidity risk.

- FINREP is the financial information reporting framework with which all European credit institutions must comply. It aims to harmonise the supervisory reporting requirements across the euro area. The templates that have to be produced cover: balance sheet and income statement; comprehensive income and equity; disclosure of financial assets and liabilities; disclosure of derivatives; and general breakdown of all assets by country and sector.

Templates should be submitted by the reporting entities with a frequency that depends on the nature of the module itself. Another important feature related to entities and having an impact on the modules is the scope of prudential reporting, which clarifies the consolidation of reporting for each entity. Under the Capital Requirements Regulation, banks are requested to comply with prudential requirements and provide the associated reporting at the individual (Ind) and/or consolidated (Con) level. In consolidated reporting, the reported values are an aggregate which also includes all the subsidiaries (entities owned by the bank). Some of the entities are required to report at both Ind and Con levels.

---

4 Commission Implementing Regulation (EU) No 680/2014 of 16 April 2014 laying down implementing technical standards with regard to supervisory reporting of institutions according to Regulation (EU) No 575/2013 of the European Parliament and of the Council (OJ L 191, 28.6.2014, p. 1).

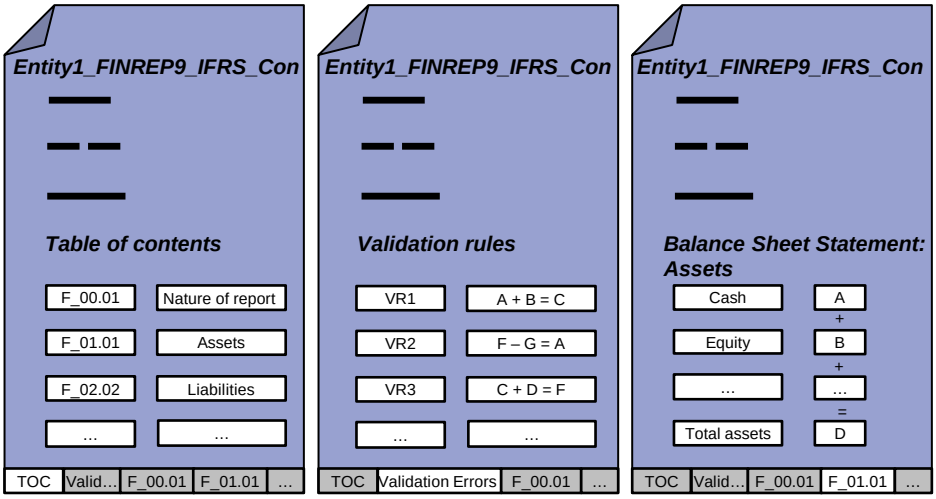
5 Regulation (EU) No 575/2013 of the European Parliament and of the Council of 26 June 2013 on prudential requirements for credit institutions and investment firms and amending Regulation (EU) No 648/2012 (OJ L 176, 27.6.2013, p. 1).

Templates can be accessed by the JSTs in the form of spreadsheet files. These spreadsheet files are organised into several sheets and follow a consistent structure. We provide an illustration of this structure below to give a high-level understanding of how information is ordered in the templates. The first sheet contains a table of contents, while the second sheet lists the validation rules that the data does not satisfy, if any. The actual data submitted by the relevant institution are contained in the subsequent sheets.

Figure 1 illustrates the schematic structure of an example FINREP template as it appears to a JST member. Following the overview in the table of contents to the left, the middle sheet illustrates the validation rules. In this illustration we use A, B, C, etc. to indicate specific reported values. To the right, you can see one of the content sheets, in this case sheet F_01.01, “Balance Sheet Statement: Assets”, listing the corresponding values.

Figure 1

Schematic illustration of a FINREP template



Data points represent the business concepts; they are the most detailed information. A template contains several data points. The value of a data point can be linked to other data points. The relationships that should hold are represented by validation rules, which are formulae or expressions that determine whether the value of a given data point is acceptable with respect to the value of the other data points. In order to ensure a uniform implementation of the ITS on Supervisory Reporting, the EBA provides a data point model (DPM) and an XBRL (eXtensible Business Reporting Language)[6] taxonomy. The DPM is a dictionary which identifies the content of each data point. In addition, in combination with the taxonomy, it defines all the business concepts and relationships, as well as validation rules. Dictionaries of reportable information are defined by means of XBRL taxonomies. XBRL is the format chosen by the EBA for reporting supervisory data. Reporting entities send their data through XBRL files, documents which follow the format defined by the taxonomy. Each data

6 See https://www.xbrl.org/the-standard/what/the-standard-for-reporting/

point is uniquely identified by a “context” and a “metric”, which together define the business concept that is represented by the data point.

To clarify how the business concept is embedded in the definition, we can take a closer look at one data point that will also be used for normalisation purposes in our approach. “Total assets” is represented by the XBRL metric XBRL:MET(EBA:mi53) and the context EBA:BAS(x6)EBA:MCY(x25), where:

- mi53 stands for “Carrying amount” monetary [m] stock [i];
- BAS defines the basic conceptual meaning of a data point and identifies the framework in which a data point is included, while “x6” indicates more specifically that the value refers to “Assets”;
- MCY specifies the concept behind the data point reported, while the value “x25” refers to “Total assets”.

In this paper, the data points will correspond to the variables of a machine learning model. The observations we use for training a model are the values reported in the templates which are characterised by the entity ID of the reporting agent and the reference period.

2.2 Quality assurance for supervisory data

As mentioned in the previous section, supervisory data are submitted to the ECB in XBRL format via the SUBA system. In order to assess the correctness, completeness and consistency of the data, several checks are performed by means of the following two types of validation rules.

- Technical: The first type of automatic checks aims to ensure that the general format of the XBRL file is correct. For example, checks are made to ensure that the name of the file follows the naming convention and that the structure of the file itself is correct. These syntactical checks ensure the formal validity of the data submitted and that the submission can be processed by the system.
- Business: Once a file has passed the technical checks, it is further assessed to ensure the quality of the ITS data itself. The checks at this stage are of semantic nature and are designed to ensure the completeness and consistency of the data with regard to business logic. Most of these validation rules are defined by the EBA and implemented through the EBA XBRL taxonomy. However, the ECB collaborates with NCAs on defining additional plausibility checks to extend and refine the data quality framework and to improve data quality across the SSM.
- The SUBA system is configured to check automatically all ITS modules received through the trigger of the validation rules. Some further checks for SI institutions are manually triggered by business experts after receipt of the data.

This paper aims to provide a further tool to support the development of new checks. These checks extend the validation rules with the objective of automatically detecting potential quality issues which might require further investigation.

3 Related work

The use of algorithms to detect patterns in data and thus help experts to gain new insights into the data has a long tradition and lies at the intersection between statistics, machine learning and data mining. Accordingly, there is a vast field of literature addressing this task and related topics.

One aspect of our work is the generation of rules based on data observations. Rule mining has been investigated for decades and in different forms. A common application is the identification of association rules (Agrawal, Imieliński and Swami, 1993 and Agrawal and Srikant, 1994). Approaches based on association rules involve looking at items found together in sets, e.g. the contents of shopping baskets, to identify patterns of items that commonly appear together. Extensions of these approaches can be used for classification rules (Han, Cai and Cercone, 1992) or to incorporate elements of explainability and interpretability (Cano, Zafra and Ventura, 2013). However, in general, these approaches operate on finite sets of discrete elements, which are different from the mainly numeric data we deal with in supervisory reporting templates.

A key technique we use for our pattern detection is regression. Many classical approaches exist for linear regression (Tibshirani, 1996, Hoerl, 1962 and Zou and Hastie, 2005). Modern regression approaches have the capacity to model more complex functional relationships in the data and are based on support vector machines (SVMs) (Drucker et al., 1997), ensemble methods (Breiman, 2001 and Geurts, Ernst and Wehenkel, 2006) or neural networks (Lathuilière et al., 2019).

Another relevant field is the general topic of outlier and anomaly detection. Several surveys provide a good overview of the relevant work and categorise approaches based on the type of method used (Hodge and Austin, 2004) or on the nature of the data, the expected outliers and the applications (Chandola, Banerjee and Kumar, 2009). Others investigate recent trends such as deep learning (Chalapathy and Chawla, 2019). Outlier detection for linear models is a specific sub-problem. There are approaches which seek to identify outliers for the purpose of excluding them from the training set for linear regression models (Fischler and Bolles, 1981) and other approaches making use of regression models to detect outliers (Benatti, 2019) or to select robust features (Tolvi, 2004).

4 Modelling the discovery of plausibility checks as a machine learning task

Validation rules ensure the integrity of the data on a business-defined semantic level. Plausibility checks serve a similar purpose and confirm that a value makes sense in the context of its observation. They are motivated by and designed on the basis of experts' experience, business models and past observations. In this section, we explain that the discovery of novel plausibility checks can be formulated as a machine learning task. We also explain how to incorporate prior knowledge into the task to ensure the novelty of the discoveries and how to eventually serve the business need of defining rules and implementing quality assurance measures.

4.1 Using regression models to discover hidden patterns in the data

As stated above, plausibility checks consider values in context and confirm whether an observed value makes sense. Context can come in different forms, such as temporal context, spatial context or the context of peer groups. The hypothesis underlying our approach is that context is defined sufficiently well by the entirety of all information reported for a given reporting agent, a given reporting period and a given consolidation level. After all, this information provides the basis for the analysis of the experts when it comes to supervisory tasks.

Hence, let us assume $X = (X_1, X_2, X_3, \cdots, X_n)$ to be the data points of a report we consider in our analysis. In terms of machine learning models, the data points correspond to variables. We can then define the context of a variable X_i as consisting of all other variables $(X_1, X_2, \cdots, X_{i-1}, X_{i+1}, \cdots, X_n)$. For a particular observation $\hat{x}$ of the variables, this means that the concrete context for an observation of the value $\hat{x}_i$ is given by the values $(\hat{x}_1, \hat{x}_2, \cdots, \hat{x}_{i-1}, \hat{x}_{i+1}, \cdots, \hat{x}_n)$.

Furthermore, we assume that a plausibility check is based on an interdependence between the variables. This interdependence might be explicitly known (e.g. because a variable represents an aggregate of other variables), or it can be implicitly present in the data and reflect features of the data-generating processes (e.g. the business model, operational targets or business processes of a supervised institution). It is these latter, implicit and unknown interdependencies we are interested in. The motivation for using machine learning is to detect such patterns in the data in a scalable and automated way and describe them using a formal model.

The vast majority of data collected for supervisory purposes are of a numerical nature, and we focus on this type of data[7]. This allows us to formulate the task of

[7] The approach can easily be extended to categorical or even textual data, for which corresponding machine learning models exist.

finding a plausibility check for variable X_i as the task of identifying a function f_i, which describes its dependence on its context:

$$X_i = f_i(X_1, X_2, \cdots, X_{i-1}, X_{i+1}, \cdots, X_n) + \varepsilon_i$$

where ε_i is an error term to ascribe a certain amount of flexibility or tolerance to the plausibility check.

Viewing the task in this way, it becomes obvious that looking for plausibility checks can be formulated as a task of performing a multitude of regression analyses: one for each variable.

The family of regression models we consider when looking for functions f_i defines the search space of functional dependencies we can identify. For instance, if we consider only linear regression models, we will only find linear dependencies. If, instead, we allow for more complex regression models, e.g. polynomial models, we might identify more complex dependencies. Hence, one of the main parameters for our approach is the decision on which types of regression models to consider.

It is important to note that we are interested in finding novel and previously unknown checks for supervisory data. This means that we try to identify novel patterns. At the same time, the large amount of predefined EBA validation rules corresponds to already known patterns in the data. Moreover, we can represent these validation rules as functions. To comply with our claim of detecting novel rules, we need to make sure we do not rediscover the already known rules using our data-driven machine learning approach.

To this end, we need to exclude all functional dependencies modelled in the validation rules from our search space. Formally, if we define E to be the set of functions representing validation rules (e.g. assume a function $e_k \in E$ with $X_a = e_k(X_b, X_c) = X_b + X_c$ to indicate that a variable X_a corresponds to the sum of two other variables X_b and X_c), then we need to make sure that we do not identify a function f which is already member of E. Let us refer to E as the set of already known validation functions[8].

The approach we take here is to constrain the search space for regression functions by excluding all functions with the same input space of variables as already known functions from the validation family. This means that if for a variable X_a we know there is rule e_k with input variables X_b and X_c then we will not look for any regression model trying to predict X_a incorporating X_b or X_c as input. Hence, X_b and X_c are entirely taken out of the search space and the prediction of X_a cannot use any information from either of the variables.

Formally, let $in(e_k)$ be the set of variables which are defined as input variables for a function $e_k \in E$, and $scope(e_k)$ be the set of variables affected by the validation rule represented by e_k. In the above example of e_k defined as $X_a = X_b + X_c$ this corresponds to $in(e_k) = \{X_b, X_c\}$ and $scope(e_k) = \{X_a\}$. Then we constrain our

[8] Note that the same validation rule might define multiple representations of the same functional dependency. For instance, our example $X_a = X_b + X_c$ is equivalent to $X_c = X_a - X_b$ and $X_b = X_a - X_c$. The set E shall contain all representations.

search for regression models to consider as input to the regression only those variables which are not already part of the input for a validation rule[9]. Hence, we are looking for:

$$X_i = f_i\left(X_j : X_j \notin \bigcup_{e_k:\, X_i \in scope(e_k)} in(e_k)\right) + \varepsilon_i$$

Formulated in a different way: we exclude all variables from the regression analysis for variable X_i, for which we already know that they are part of a functional dependency according to a validation rule. In this way our approach is forced to find new dependencies and, as a result, novel rules.

Remark: If the variables constituting a functional dependence stemming from validation rules are not excluded, our approach will be strongly biased towards identifying exactly such rules. This can be understood relatively easily as follows.

The EBA validation rules are used to check the quality of the supervisory data submitted. Hence, the data in the SUBA system will (broadly) comply with these rules. This means that, for instance, the validation rule $X_a = X_b + X_c$ will also cause the observed values of variable X_a to be exactly the sum of X_b and X_c. When training a regression model on this data, the solution of finding exactly the function $X_a = X_b + X_c$ will immediately be an optimal one, as it leads to a minimal error term for the model on the training data. Hence, the detected rules will conform to the already known EBA validation rules.

In fact, in an initial stage we applied our method without the exclusion of known dependencies stemming from the EBA validation rules. As expected, we identified perfect prediction models for all variables involved in validation rules. This initial application of the approach was used as a sanity check to verify the correctness of our implementation.

4.2 Obtaining business-oriented checks from the results of a regression analysis

We mentioned above how we can interpret the task of finding plausibility checks as a multitude of regression analyses. Correspondingly, the result of our search for new plausibility checks is a collection of regression models. Considering those regression models, there are two ways they can support the actual task and business logic, i.e. to improve data quality assurance based on new rules.

(a) **Interpretation of the regression models themselves:** The regression models can provide direct motivation for the introduction of additional checks. Provided the models come with a sufficient degree of interpretability, experts can take a look at the functions of the regression models. This permits business experts to validate the identified functions

[9] There might be multiple validation rules affecting the same variable, i.e. with the same scope.

against their domain expertise and background knowledge. Once they have made sure that the business logic is sound, the business experts can define new plausibility checks in SUBA. Effectively the outcome is a function in the same format as the EBA validation rule specifications. The advantage of this approach is that it allows for the identification of very generic rules which have been backed by business logic. In a certain way, it can be seen as a tool for inspiring and guiding the experts in the development of domain-driven rule definitions.

(b) **Direct identification of implausible values:** The regression models can also be applied directly to detect implausible values. To this end, any observed variable value can be compared with the value predicted by the regression model, given the context of all other variables. If the observation deviates too much from the predicted values, this is a good indication of an anomaly and might require further investigation. The advantage of this approach is that it does not necessarily require the regression models to be interpretable. This means that the search space for identifying functional relationships between the variables can be larger and include more complex functions.

In this paper we investigate both approaches for making use of the learned regression models. In particular, we introduce a normalised way of measuring deviations from the predictions which allows for a harmonised assessment, even for variables with very different value ranges.

5 Implementation

In this section we go into the details of the implementation of our approach. We illustrate how we prepared, filtered and transformed data from SUBA to render it suitable for our analysis, how we technically incorporated prior knowledge on existing validation rules and how we built our regression models using different machine learning techniques.

5.1 Data preparation

For the purpose of this study we focus on FINREP, as this framework contains clear relationships among the data points. In addition, the framework was changed at the end of 2018 to comply with IFRS 9 standards introducing new concepts and templates, so it is a good candidate for checking the effectiveness of the model in identifying new patterns. FINREP modules are reported on a quarterly basis, and we considered all available consolidation levels, both Ind and Con.

The selection of templates[10] was driven on the one hand by the presence of several validation rules for these templates, which served as prior knowledge to be considered in the process, and on the other hand by the suspected potential for latent relationships to be discovered[11].

Data are stored in SUBA as a list of key-value pairs (data point, value). This representation is optimal for sparse data. However, most machine learning frameworks need data to be represented as a matrix. To match this requirement, we reformatted the data into a matrix. In this new matrix, each column corresponds to a data point and each row to an observation for an entity at a given reference period.

Given the initial sparsity of the data, many of the values in the matrix are empty, as they were not reported or not used. We handled the missing data applying the following, business-motivated guidelines.

- Fill missing data with zeros for templates that have been reported for a specific entity and reference period (we know that a template has been reported for an entity and a reference period if there are data for at least one of the data points defined in this template). This corresponds to the business interpretation in a case where the value is assumed to be reported but also assumed to be zero.
- Leave missing data empty for those variables for which this template has not been reported. In this case we assume that the entity did not have to report the

[10] FINREP templates containing information on breakdown of financial assets and non-performing exposures.

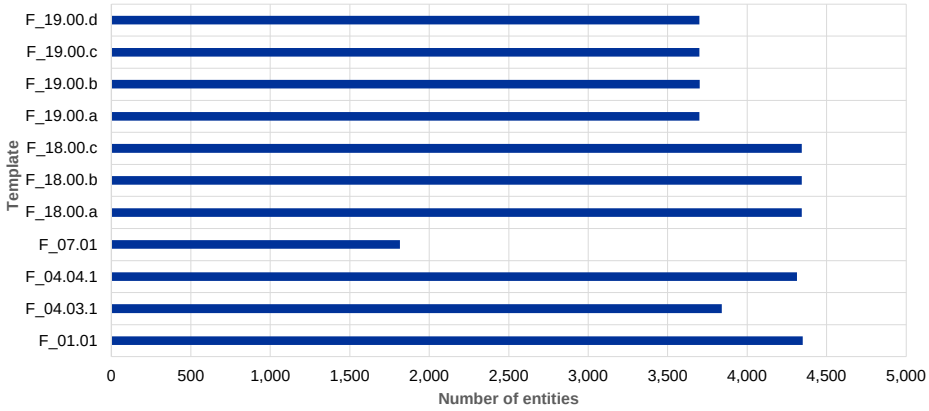
[11] As a first assessment, we verified that the model actually detects the trivial validation rules. In a second phase we excluded the already known relationships to gain new insights. In this paper, we focus on the detection of new insights.

template, and the data are intentionally missing (which is a different signal than a value of zero).

According to this treatment of missing values, we investigated whether there were sufficient data for a systematic analysis. Chart 1 depicts how many observations were reported for entities at different reference periods in the templates considered. We noticed that one template (F_07.01) was reported on far fewer occasions: this template comprises breakdowns of financial assets subject to impairment and is not required to be reported by some less significant institutions. As we expect novel plausibility checks to appear mainly across templates, we decided to exclude the variables from this template for this analysis. Keeping F_07.01 in this analysis would have limited our analysis to far fewer observations.

Chart 1

Amount of entities per reported period for which specific template data are available



After this first preparatory step, we were left with 2,619 variables to consider. However, due to the sparse nature of the reported data, very few variables have data all the time. This lack of data can be problematic when trying to use machine learning to automatically learn relationships. To overcome this obstacle, we took the following two decisions.

- We considered only those data points that contained data for at least 5% of the observations. In this way the initial list of 2,619 variables was reduced further to 942 usable data points.
- When training the machine learning models for variable X_i (e.g. $X_i = f_i(X_1, \cdots, X_{i-1}, X_{i+1}, \cdots, X_n)$) we considered only the observations for which values for X_i were actually reported. In practice this corresponded to excluding the prediction of missing values from the models.

The next step in the pre-processing pipeline is data normalisation. In machine learning, data are usually normalised so that each attribute is in a common and predefined range (e.g. [0, 1]) or of similar scale (e.g. by standardising its values). Otherwise, optimisation techniques, such as gradient descent, or distance-based

methods (e.g. clustering) might give higher importance to the variables with higher magnitude[12].

However, there is yet another reason to normalise data: to overcome certain aspects of sparsity in the feature space. According to Bengio, Courville and Vincent (2013) a good representation of data is insensitive to small variations in the values and exhibits a spatial coherence. If data observations are represented in a dense way, i.e. close to each other, the machine learning models learned from this data tend to be more representative and generalise better. In the era of big data, this is usually not a concern, as big data typically implies dense data. In our case, due to a change in the accounting framework, we only had data from three reporting periods to hand, namely the first, second and third quarters of 2019. Involving data from previous periods would have incorporated a systematic bias into the data for those values which were affected by the change in accounting standards. Due to this limited amount of data, we preferred to include normalisation techniques that contribute to a denser representation.

We considered the following approaches for normalisation:

- **Total assets**: This normalisation divides every variable by the total assets reported for the corresponding entity. The main advantage is that it is a business rule (intuitive in this domain) in which each variable is expressed as percentage of total assets. The disadvantage is that it leads to numerical precision issues (e.g. many variables are very close to 0 in the normalised space). This in turns makes it almost impossible for machine learning models to make use of close-to-zero values.

- **Million**: The million normalisation divides every variable by 1 million. This normalisation solves the numerical stability problem of total assets normalisation, as typically the values are neither too small nor too big. However, the representation is still sparse when we take into account the amount of data available.

- **Quantised**: The quantised normalisation transforms each variable in a quantile in the interval [0, 99]. The quantisation is performed by assigning an equal number of values to each quantile for each variable. This normalisation preserves the order of observations, in the sense that higher (lower) values in the original space correspond to higher (lower) quantiles. It also has the advantage that it is robust against outliers and anomalies in the data. Extremely high or low values are grouped together with other values and represented by the corresponding quantile. Hence, the absolute values of anomalies have no impact on the regression models. The quantised normalisation also solves numerical precision issues from the total assets normalisation. And finally, it makes it easy to compare the performance of machine learning models in predicting variables of very different scales, as the error can be expressed in a 100-quantile scale. The disadvantage is that the machine learning models will

[12] Not every machine learning technique requires data to be normalised. For instance, tree-based techniques, such as random forests or extremely randomised trees, also work well on non-normalised data.

provide predictions in the quantile space, which is harder to interpret from a business perspective. To overcome this limitation, we can transform this prediction back into the range of values in the original space. One aspect to consider is that the quantised normalisation performs a non-linear transformation on each variable in the data. Overall, this corresponds to a non-trivial transformation of the analysed data and might obscure some simple linear relationships in the data. This effect needs to be taken into account when selecting suitable regression methods (cf. Section 5.3).

Table 1 summarises the advantages and disadvantages of each normalisation technique.

We will report only on the results making use of quantised normalisation, as its advantages are very relevant to this problem, while its disadvantages can be easily overcome in practice by providing an approximate inverse transformation.

Table 1
Overview of advantages and disadvantages of the normalisation techniques considered

Normalisations	Advantages	Disadvantages
Total assets	Business rule (% total assets)	Numerical precision
Million	Solves numerical precision	Sparse representation
Quantised	Solves numerical precision Easy to compare predictability Robustness to anomalies	Prediction in quantised space

5.2 Handling prior knowledge

Since in this project we are looking at the possibility of finding novel relationships that are useful for performing plausibility checks, we need to handle prior knowledge and prevent machine learning models from exploiting the already known relationships. As described in Section 4.1, our solution is to exclude the variables modelled in prior knowledge from the input that the machine learning models can use. This forces the machine learning models to learn alternative and non-trivial relationships not covered by prior knowledge.

For example, given the known rule $a = b + c$, neither the variable b nor the variable c must be considered as an input variable to predict a. We use the following notation to denote which variables must not be considered as input variables: $\mathrm{exclude}[a] = \{b, c\}$. Note that it is not important to keep track of the type of relationship between the variables (e.g. an addition in this example). Therefore, the prior knowledge of any rule can be simplified and formulated as $a \sim b,$ and $a \sim c$ with the meaning that a depends on b and c.

Note that this also implies that b depends on a and c. This can easily be seen in the above example, as we can reformulate the initial rule into $b = a - c$. Likewise, c depends on a and b. In fact, our interpretation of dependencies in the known rules

implies a symmetric and transitive relationship. Thus, we can directly deduce from $a \sim b$ and $a \sim c$ that also $b \sim c$.

This observation indicates that excluding only variables participating in directly known relationships is not sufficient. For instance, assume the variable b in the above example appears in a further rule with variable d, indicating $b \sim d$. Based on the transitivity of $a \sim b$ and $b \sim d$ we see that $a \sim d$. Hence, we also need to exclude d from the input variables used in the prediction model for a.

This means that we also need to take care of variables involved indirectly in relationships. In order to prevent machine learning from making use of prior knowledge, we need to implement the transitive closure of dependencies. By "transitive closure" we mean the need to expand dependencies as far as possible.

Table 2 shows an extended example considering three prior knowledge rules: $a = b + c$, $b = 2 \cdot d$ and $e = a - f$. The rule $a = b + c$ directly indicates the dependencies $a \sim b$ and $a \sim c$, rule $b = 2 \cdot d$ implies $b \sim d$ and rule $e = a - f$ implies $e \sim a$ and $e \sim f$. In the initial iteration, this can directly be translated into $\text{exclude}[a] = \{b, c, e\}$, $\text{exclude}[b] = \{a, d\}$, etc. by considering for each variable the dependency relationships it appears in. However, since $\text{exclude}[b]$ contains an a, this set needs to be expanded, as a itself also depends on other variables. As a result, the second iteration for $\text{exclude}[b]$ contains $\{a, d, c, e\}$. The same process also has to be applied for the other variables. This expansion can be done iteratively until no further changes appear. Table 2 shows the complete iterative expansion for the example of the three hypothetical rules.

Table 2
Transitive closure of dependencies for three hypothetical rules

a ~ b, a ~ c b ~ d e ~ a, e ~ f	1st iteration	2nd iteration	3rd iteration
exclude[a]	{b, c, e}	{b, c, e, *d, f*}	{b, c, e, d, f}
exclude[b]	{a, d}	{a, d, *c, e*}	{a, d, c, e, *f*}
exclude[c]	{a}	{a, *b, e*}	{a, b, e, *d, f*}
exclude[d]	{b}	{b, *a*}	{b, a, *d, e, f*}
exclude[e]	{a, f}	{a, f, *b, c*}	{a, f, b, c, *d*}
exclude[f]	{e}	{e, a, *b, c*}	{e, a, b, c, *d*}

In our final solution, each of the machine learning models uses all the available variables except those that are part of its transitive closure of dependencies. With the EBA rules applicable to the templates we considered in our experiments, five iterations were sufficient for the transitive dependencies to be made completely explicit.

5.3 Machine learning models

The approach we take is to predict which value each variable should have as a function of other variables. We then define a score that takes into account their discrepancy to signal implausible values. In this section, we discuss which machine learning regression models we have considered to make predictions and the trade-off decisions we have taken. Given the two possible business use cases for the learned models, i.e. providing inspiration for new plausibility checks and support in detecting anomalous or suspicious values, we considered different characteristics of potential approaches. Overall, we decided to use the following criteria to select the most appropriate machine learning model.

- **Modelling capability**: The ability of the model to learn novel relationships between transitive independent input variables and the output variable. This also includes the complexity of the relationships a model is able to describe.
- **Generalisation**: The ability of the learned machine learning model to perform well on new data (e.g. data not used to train the model).
- **Amount of data needed**: Minimum amount of data needed to obtain machine learning models that generalise well.
- **Explainability**: How easy it is to interpret the way in which the machine learning model is making predictions.

The prioritisation given by the overall business use cases indicated that the first priority would be to automatically detect implausible values. Understanding the reasons why machine learning models suggested a value to be implausible was the second priority.

The machine learning models that we considered were: linear regression (e.g. lasso, ridge, elastic net) (Friedman, Hastie and Tibshirani, 2010 and Rifkin and Lippert, 2007), tree-based methods (decision trees, random forests, extremely randomised trees) (Breiman, Friedman and Stone, 1984, Breiman, 2001 and Geurts, Ernst and Wehenkel, 2006) and neural networks (Rumelhart, Hinton and Williams, 1986). All models were used in the implementations provided by the scikit-learn library (Pedregosa et al., 2011).

In the context of the choice of methods, it is worth noting that our approach is based on the assumption of latent relationships in the data. Such relationships might be of a linear nature and imply that there might be cases of collinearity in the input variables. The business rules we covered in the exclusion of prior knowledge are good examples of such relationships. The analysis of individual input variables in linear regression models can suffer from collinearity in the data. In practice, this is commonly treated with an additional step of pre-processing to detect and address collinearity. However, in our case such a treatment might obfuscate relationships in the data and would thus be counterproductive. Furthermore, ensembles of randomised methods such as random forests and extremely randomised trees are

robust to collinearity. Therefore, we intentionally did not treat collinearity in the input data beyond the exclusion of business rules.

5.3.1 Linear regression

Linear regression is a simple machine learning algorithm that can be used to model linear relationships. It is fast to train and also works well with small amounts of data. Its main limitation lies in its modelling capability, as it can only learn linear relationships. If the relationship is linear, it generalises very well. Otherwise, it generalises poorly. Its major strength is its explainability: the coefficients of the trained model provide very detailed information on how the predictions are done and which variables have the most influence.

Linear regression comes in several flavours depending on the regularisation used (e.g. how the coefficients are penalised in the training process in order to improve generalisation).

- **Lasso** (Tibshirani, 1996) is a good option when we suspect that the majority of the variables will not be useful for making predictions. This is a reasonable assumption in our case, as we suspect that only a few variables will be necessary to predict another. Lasso uses L1 regularisation.
- **Ridge** (Hoerl, 1962) is useful when we suspect that the variables might be correlated. Ridge uses L2 regularisation.
- **Elastic net** (Zou and Hastie, 2005) combines the advantages of both lasso and ridge.

5.3.2 Tree-based methods

Tree-based methods are non-parametric machine learning techniques that work well in many circumstances as they can model any kind of behaviour. They need more data than linear models but do not require big amounts of data. Tree-based methods can provide information on the relative importance of each of the input variables. However, these results can be misleading if there is collinearity in the selected features. We considered three tree-based solutions.

- **Decision trees** (Breiman, Friedman and Stone, 1984, Quinlan, 1983 and Quinlan, 1986) is the name given to a solution consisting of a single tree. It is very explainable, as the tree structure can be inspected and a user can understand what the decisions are based on. The problem with decision trees is that they are very biased (e.g. they have a good training performance and often a poor generalisation performance).
- **Random forests** (Breiman, 2001) are bagging ensembles of decision trees. This means that different trees are trained in different portions of data with possibly different features. The final prediction is made by aggregating the

predictions from each tree (e.g. using their average). Random forests also provide information about the feature importance. The problem with the feature importance of the random forests (and the trees) is that it is biased. This is just because of the way the trees are built. The threshold used to take a decision in every branch is made to optimise the split. This results in features with a wider range being overrepresented almost by pure chance, as higher range features will increase the chances of finding a decision threshold that will split the data better.

- **Extremely randomised trees** (Geurts, Ernst and Wehenkel, 2006) overcome the problem of this additional bias from random forests by making decision splits randomly. This allows us to have a better understanding of the true feature importance. In addition, extremely randomised trees are faster to train as they do not need to find the optimal splitting point.

5.3.3 Neural networks

Neural networks are a machine learning technique inspired by the brain (Rosenblatt, 1958 and Ivakhnenko, 1973), and its neurons in particular. They are able to model complex non-linear relationships, and their generalisation capability is very good (Schmidhuber, 1992 and Salakhutdinov, Mnih and Hinton, 2007). Their main limitation is that they typically need much more training data than other classical machine learning algorithms in order to outperform them. Another disadvantage is the lack of interpretability. It is difficult to understand looking at the trained neural network weights what the predictions are based on.

5.3.4 Trade-off

In order to decide which machine learning technique would work best for this project, we scored each technique on each of our criteria. We considered specific aspects of our problem such as how much data was available for training the machine learning models. Table 3 shows the individual scores for each technique and indicates that **extremely randomised trees** are the most promising machine learning technique in our case.

Table 3

Individual scores for each machine learning technique considered

Score	Capability	Generalisation	Amount of data	Explainability	Average score
Linear regression	1	1	5	3	2.5
Elastic net	1	2	5	5	3.25
Decision trees	4	2	4	5	3.75
Random forests	4	4	4	3	3.75
Extremely randomised trees	4	4	4	4	**4.0**
Neural networks	5	5	1	1	3.0

Note: The lack of explainability of neural networks could be compensated for by the use of modern explainability frameworks such as SHapley Additive exPlanations (SHAP) (Lundberg and Lee, 2017), local interpretable model-agnostic explanations (LIME) (Ribeiro, Singh and Guestrin, 2016) or LICON (Kasneci and Gottron, 2016). However, the amount of data available in our project was considered insufficient for neural networks to be effective.

5.4 Training the machine learning models

As stated before, we worked with data from the first, second and third quarters of 2019. We decided to use the first and second quarters for training the machine learning models and reserved the third quarter as an evaluation set to look for implausible values (see also Section 6)[13]. This time-oriented split of the data was chosen because of the use case and the need to check the validity of the models from one reporting period to the next. From a business point of view we did not expect a high volatility on a quarter to quarter basis for the set of variables used. In addition, this split of data satisfied the methodological requirement to have a clean separation of training and test data. In particular, we wanted to ensure that there was no information leakage between the training and test datasets. Using a split along the time dimension of the dataset ensures this clear separation. It excludes the possibility of observations stemming from the same instance of a reporting template to be used for both training and testing.

The training was performed on data using quantised normalisation as discussed in Section 5.1. This implies that the predictions on new data were also provided in the quantised space. As discussed in Section 5.3.4, we used extremely randomised trees as the machine learning technique to model the relationships between variables. We trained a total of 942 extremely randomised trees regression models in the form:

$$X_i = f_i(X_1, \cdots, X_{i-1}, X_{i+1}, \cdots, X_n)$$

We made sure we excluded prior knowledge by excluding the transitive closure of dependent variables from the machine learning model input. In addition, we only considered as training data the instances for which X_i had been reported in its corresponding financial template.

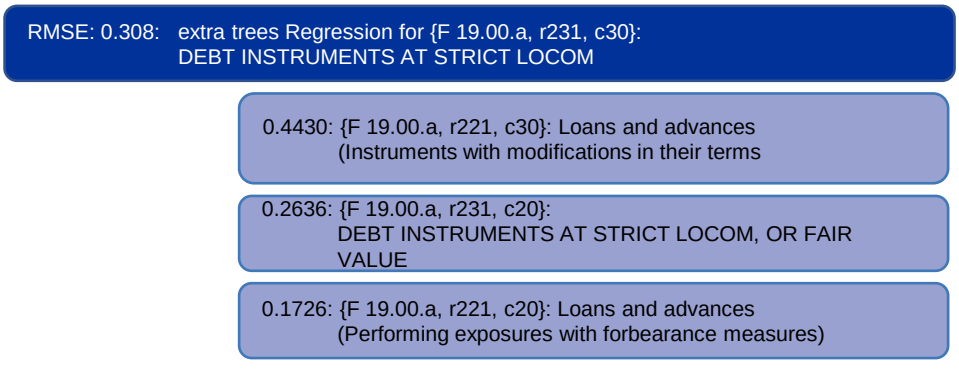
[13] We did not explicitly test how robust the model is to new entities. The behaviour of the model to completely new entities might be reviewed in future work.

When training and evaluating a regression model for a specific variable X_i, we obtain several information and insights. First of all, we obtain the model itself and, as described above, a notion of feature importance. The model and the ranked relative feature importance provide the basis for an expert's decision to implement new plausibility checks. They indicate which variables, i.e. data points played an important role in predicting the values of variable X_i. This is the type of guidance or inspiration we seek to give to business experts for defining new plausibility checks. Second, using the evaluation data makes it possible to test the model's ability to generalise. By comparing the predictions from the trained models with the actual values of the training data, we will typically observe some deviations. The root mean squared error (RMSE) is a standard measure for such deviations. The higher the value, the less precise the predictions are. Low RMSE values therefore indicate very good predictive quality for the model on the evaluation data, which in turn indicates good generalisation.

Overall, we condensed the information from the training and evaluation phase into the above key characteristics, i.e. we indicated the data point a model was trying to predict, how well the prediction matched the observed values in the evaluation data in terms of RMSE and which other data points were most influential in the prediction. An example of this output indicating the quality of the model and the information providing explainability can be seen in Figure 2. This information provided the input for the evaluation of the approach by business experts, which will be discussed in Section 6.

Figure 2

Illustrative example of an extremely randomised trees regression model, with RMSE measured in quantiles



5.5 Implausibility score

While the trained models as presented in the previous section provide insights in terms of quality of the models and the important underlying features, they do not directly help in assessing specific observations to determine whether they are outliers or implausible values.

In order to find out which values (e.g. observed data points) seem implausible, we used an anomaly detection approach. The idea of this approach is to measure, in a normalised way, how much an observation in the evaluation data has deviated from its predicted value. This normalised implausibility score can then be used to rank the observations starting from the most extreme deviation from the predicted value. Ranking observations in this way helps the data quality managers to focus their attention on the most important data points and efficiently identify outliers.

The approach for computing the implausibility score is summarised by the following high-level steps.

1. For each financial institution, each variable is predicted as a function of all other variables (in the same financial institution) with the models trained as described in Section 5.4.

2. The models are additionally evaluated with regard to their performance on the training data. This does not help in assessing their ability to generalise but provides an idea of how much variance or noise there is in the data. Specifically, this makes it possible to assess how much deviation from a predicted value can be expected.

3. The machine learning predictions for an observation are compared with the actual values. The deviation is measured in quantised space and normalised by the experienced deviations during the training of a model. This normalised deviation will be expressed as an *implausibility score.*

4. A list of implausible values (sorted by higher score) is generated and provided to the data quality managers for their expert assessment.

There are several options for computing the normalised implausibility scores. Our goal is to minimise the number of false positives, i.e. to ensure that the top ranks of the list of implausible values actually correspond to anomalies. In other words, our objective is to maximise the chances of being right when declaring a value as implausible. In machine learning terms, we prefer precision to recall. This means that the implausibility score not only needs to incorporate how much an observed value deviates from the predicted value but also needs to comprise a notion of how likely it is to observe such a deviation.

In order to minimise the number of false detections, we acknowledge that the trained machine learning models are not perfect and we can therefore expect some errors in their prediction. In our approach, a value is potentially implausible if its prediction error is outside an expected error interval. In order to characterise the expected error interval, we consider the prediction error in quantised space in the training set as depicted in Figure 4. If the deviation between prediction and observed value is zero, this corresponds to a prediction which lies within in the same quantile. If the deviation amounts to a value of 3, for instance, this means the prediction is three quantiles above the real value.

Since we cannot discount the fact that there were already implausible values in the training set, we propose using as plausible error the error interval between the 1st

and the 99th percentiles[14] of the prediction error in quantised space in the training set. This means we consider the observed errors of the model while training and use them for normalisation. This corresponds to the idea that a certain range of deviations from the predictions can be expected and is inherent in the data and the capacity of the model. To cater for asymmetric and skewed error distributions, we calculate the characterisation error separately for deviations above and below zero.

The implausibility score is computed by simultaneously considering the prediction error and the characterisation error (see equation 1). The characterisation error, as defined by the two extreme percentile thresholds on the corresponding sides, serves as a normalisation factor. The advantage of this approach is that it makes the scores of different variables comparable even if the underlying machine learning model errors are different.

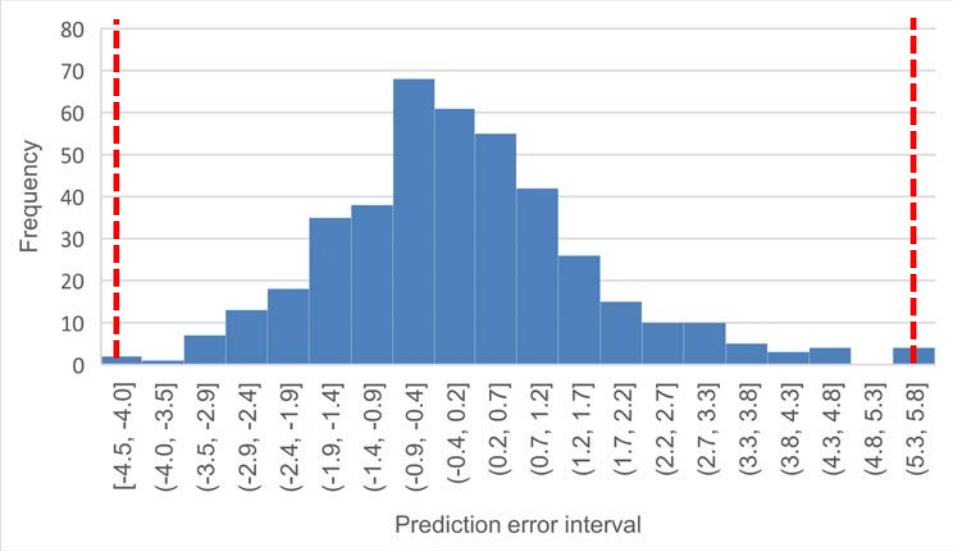
$$(1) \qquad implausibility\ score = \frac{|\text{predicted} - \text{real}|}{|\text{corresponding characterisation error}|}$$

Let us take a closer look at the example depicted in Chart 2 to illustrate how the implausibility score works. Assume we observe a prediction error of 2, i.e. the prediction is two quantiles above the actual value in the quantised space. The characterisation error of the model above zero corresponds to the last percentile. The threshold is indicated by the red line to the right at 5.7. This means for this particular model that, for only 1% of the cases in the training data, the prediction exceeded the true value by more than 5.7 in the quantised space. Normalising the observed error, we obtain an implausibility score of $\frac{|2|}{|5.7|} = 0.35$. Meanwhile, if we observe a prediction error of -10, we get a different score. As the characterisation error is negative, we now consider the first percentile of deviations observed during training. This threshold, indicated by the red line to the left, corresponds to a value of -4.3 in the quantised space. The implausibility score in this case is $\frac{|-10|}{|-4.3|} = 2.33$.

[14] The exact thresholds are not important (e.g. using the 2nd and 98th percentiles instead would work just as well) as they are only used as normalisation factors.

Chart 2

Prediction error in the quantised space in the training set for one of the variables



Note: The red lines correspond to the first and last percentiles.

Note that with this approach we compute implausibility scores for each data point value in a reported template. To assess whether the overall template corresponds to an anomalous observation, it is sufficient to aggregate the scores of all data point values it contains.

6 Experimental set-up, results and evaluation

We performed two types of qualitative evaluation of our approach. The evaluation was geared towards testing whether the approach helps in the two tasks we identified, namely (a) the discovery and definition of new plausibility checks and (b) finding particular values which constitute outliers and are worth further investigation in a data quality assurance process.

6.1 Discovery of new plausibility checks

The first part of the evaluation focused on the ability of our approach to support business experts in defining new plausibility checks. To this end, we used the results we obtained from training the regression models as described in Section 5.4. We selected the models that showed good predictive performance in terms of low RMSE values. We then used the predicted data points and the list of the most influential input variables from those models and showed this information to business experts for further evaluation.

The business experts investigated ten patterns identified by the machine learning models in the data. These ten patterns were selected on the basis of their predictive performance and high accuracy. They focused on the relationships across templates presenting financial assets subject to impairments in different breakdowns. All the relationships made sense from a business point of view and confirmed the results obtained in a separate thematic review, carried out independently by the business experts, on non-performing exposures reported in FINREP.

Some of the rules provided interesting new insights. They highlighted the strong relationships between templates F18 (information on non-performing exposures) and F19 (information on forborne exposures), which were known to the business experts from their thematic review. However, they revealed that there were no existing EBA-defined validation rules covering such aspects.

These insights enabled the experts to define new plausibility checks which have been implemented and are currently used to assess the quality of data and identify cases of wrong reporting.

One of these checks assumes that impaired exposures without forbearance measures are considered defaulted (with possible exceptions according to paragraph 39 of the EBA Guidelines on the application of default definition (EBA-GL-

2016-07[15]))[16]. Based on our findings and subsequent investigations, nine institutions confirmed wrong reporting for the fourth quarter of 2019.

Overall, the approach was deemed highly suitable for guiding the experts in the discovery of new plausibility checks. The combination of assessing the quality of a predictive model and identifying the most influential variables provides a clear idea of what to consider. The feature of excluding prior knowledge prevents already known rules from being rediscovered and allows business experts to focus on ideas for novel checks.

6.2 Assessments of outliers and implausible values

The second part of the evaluation was focused on concrete data quality assessment processes and on identifying anomalies for concrete data submission. To this end, we used the above definition of implausibility scores (cf. Section 5.5) and applied them to the evaluation data for the third quarter of 2019. As a result, we obtained a ranking of values which indicated in a comparable and normalised way how far observations deviated from the values as predicted by the machine learning models.

Table 4 shows an anonymised list of the highest implausible scores computed for data from the third quarter of 2019. The list includes the machine learning model's output of the predicted range for an implausible value. This value range allows domain experts to understand better why an observed value has been declared as implausible. The prediction is expressed as a range due to the quantile nature of the prediction.

15 EBA Guidelines on the application of the definition of default under Article 178 of Regulation (EU No 575/2013).

16 Syntax: IF (({F18.00.a, c110} - {F19.00.a, c090}) >= 1000000 AND ({F18.00.a, c120} -{F19.00.a, c100}) >= 1000000) THEN (({F18.00.a, c120} - {F19.00.a, c100}) / ({F18.00.a, c110} - {F19.00.a, c090}) <= 1.00).

Table 4

Top implausibility scores for the third quarter of 2019 (anonymised)

Entity ID	Module ID	Score	Variable	Predicted interval
Entity 1	FINREP9_Ind_IFRS	25.26	{F 18.00.a, r70, c80}: Loans and advances (Past due > 90 days <= 180 days)	[8959304, 10150230]
Entity 2	FINREP9_Con_IFRS	16.76	{F 19.00.b, r330, c160}: DEBT INSTRUMENTS OTHER THAN HELD FOR TRADING OR TRADING (Refinancing)	[-6808, -1881]
Entity 3	FINREP9_Con_IFRS	15.00	{F 18.00.b, r330, c190}: DEBT INSTRUMENTS OTHER THAN HELD FOR TRADING OR TRADING (Past due > 1 year < = 5 year)	[-2528117, -2032397]
Entity 4	FINREP9_Con_IFRS	14.23	{F 18.00.b, r510, c130}: Credit institutions (Accumulated impairment, accumulated negative changes in fair value due to credit risk and provisions)	[241, 1000]
Entity 1	FINREP9_Ind_IFRS	12.50	{F 18.00.a, r180, c80}: DEBT INSTRUMENTS AT COST OR AT AMORTISED COST (Past due > 90 days <= 180 days)	[1297018, 1477429]
Entity 1	FINREP9_Ind_IFRS	11.03	{F 18.00.a, r330, c80}: DEBT INSTRUMENTS other than HFT (Past due > 90 days <= 180 days)	[1831523, 2143668]
Entity 5	FINREP9_Con_IFRS	10.89	{F 19.00.a, r180, c100}: DEBT INSTRUMENTS VALUED AT COST OR AT AMORTISED COST (of which: Impaired)	[33012889, 37012762]
Entity 6	FINREP9_Con_IFRS	10.62	{F 18.00.b, r330, c190}: DEBT INSTRUMENTS OTHER THAN HELD FOR TRADING OR TRADING (Past due > 1 year < = 5 year)	[-899269, -712853]
Entity 5	FINREP9_Con_IFRS	10.60	{F 19.00.a, r330, c100}: DEBT INSTRUMENTS other than HFT (of which: Impaired)	[31562551, 35041234]
Entity 7	FINREP9_Con_IFRS	10.51	{F 18.00.b, r40, c140}: Credit institutions (Performing exposures - Accumulated impairment and provisions)	[-31045, -23384]

The ranked values were shown to business experts for assessment. The experts investigated the top 100 values identified in detail. Note that several cases corresponded to reports submitted by the same entity, which was reporting for the first time after a merger. Several cases raised by our approach corresponded to implausible values, which were flagged independently by the data quality assessment procedures of the regular production cycle for supervisory data, showing consistency between the two approaches.

In addition, the cases analysed gave rise to some additional suspicious values which were not flagged by the data quality assessment procedures. These suspicious values were subsequently prioritised: ten cases (among the top 20 cases not flagged by the standard procedure) were selected as worthy of further investigation by the experts. These ten cases were then presented to the reporting institutions for assessment. The reporting institutions confirmed the submission in most of the cases, and in one case agreed on a correction.

In conclusion, the approach demonstrated that it is well capable of identifying outliers in supervisory data. In particular, the ability to widen the context to encompass all other reported values renders the approach more flexible and independent from fixed thresholds applicable to individual data points over the single time dimension.

7 Conclusions and future work

In this paper, we have presented a data-driven approach for mining new plausibility checks for supervisory data. The approach makes use of machine learning techniques and leverages supervisory data reported in the past to detect latent patterns in the data. The approach serves business experts in two ways: (a) by inspiring them to formulate new plausibility checks which can be implemented in reporting systems and thus have permanent, positive effects on data quality and (b) by making it possible to carry out ad hoc investigations into specific observations of anomalous or suspicious values.

The approach is based on formulating the task of looking for new plausibility checks as a machine learning task that consists of solving multiple regression problems. We described the formal basis for this approach and illustrated how to incorporate prior knowledge into the process. In implementing the approach, we considered several choices of machine learning algorithm. A solution based on the extremely randomised trees algorithm provided the best results in terms of performance and explainability of the models. As an additional step, we introduced a normalised implausibility score for individual values, enabling global ranking and prioritised assessment of reported data.

In a qualitative evaluation, we worked with business experts to investigate the extent to which the approach provides a solution for the definition of new checks and the detection of suspicious values. For both use cases, the experts used the tool to investigate the findings revealed by our automated approach on a selected number of reporting templates. The results helped for both use cases, i.e. it allowed specific new plausibility checks to be defined and it enabled suspicious values requiring further investigation to be identified. Overall, the approach has shown its benefits for the use cases considered and will be used for investigative data quality management in the future.

The evaluation has also provided some insights into potential extended applications. We have already started to investigate other sets of templates for identifying new plausibility checks. Preliminary results have shown that in the context of the COREP framework, too, we were able to identify latent relationships which may serve as basis for new rules. An interesting next step will be to extend the investigations to relationships that span data from both the FINREP and COREP templates. At present, such relationships are not subject to any checks, so they represent a promising line of investigation. Another extended application regarding data is to cover more reporting periods and to look for seasonal patterns. Finally, there are some technical improvements that might be worth investigating. One main area for improving the performance of the predictive model is to make a better distinction in the data between missing values and semantic zero values. For instance, we considered differentiating between models for predicting the presence of a data point and those for predicting its actual values. This would make it easier to identify implausible values that actually represent missing or non-reported data. Another

area of work on future extensions is a broader evaluation of other models for the regression analysis, also taking into consideration ensemble methods. This might lead to further improvements in prediction quality and needs to be accompanied by explainable artificial intelligence (XAI) methods.

References

Agrawal, Rakesh and Srikant, Ramakrishnan (1994), “Fast algorithms for mining association rules”, *Proceedings of the 20th VLDB Conference,* pp. 487-499.

Agrawal, Rakesh, Imieliński, Tomasz and Swami, Arun (1993), “Mining association rules between sets of items in large databases”, *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data*, pp. 207-216.

Benatti, Nicola (2019), “A machine learning approach to outlier detection and imputation of missing data”, *IFC Bulletins*, chapter 49.

Bengio, Yoshua, Courville, Aaron and Vincent, Pascal (2013), "Representation learning: A review and new perspectives", *IEEE transactions on pattern analysis and machine intelligence*, Vol 35, No 8, pp. 1798-1828.

Breiman, Leo (2001), “Random Forests”, *Machine Learning*, No 45, pp. 5-32.

Breiman, Leo, Friedman, Jerome and Stone, Charles J. (1984), *Classification and regression trees*, CRC Press.

Cano, Alberto, Zafra, Amelia and Ventura, Sebastian (2013), “An interpretable classification rule mining algorithm”, *Information Sciences*, Vol 240, pp. 1-20.

Chalapathy, Raghavendra and Chawla, Sanjay (2019), “Deep learning for anomaly detection: A survey”, *arXiv preprint arXiv:1901.03407*.

Chandola, Varun, Banerjee, Arindom and Kumar, Vipin (2009), “Anomaly detection: A survey”, *ACM Computing Surveys (CSUR)*, Vol 41, No 3, pp. 1-58.

Detken, C. and Nymand-Andersen, Per (2013), “The new financial stability framework in Europe”, *Handbook on Systemic Risk*, pp. 748 - 774, Cambridge University Press.

Drucker, Harris, Burges, Chris J.C., Kaufman, Linda, Smola, Alexander J. and Vapnik, Vladimir N. (1997), “Support Vector Regression Machines” *Advances in Neural Information Processing Systems*, pp. 155-161.

Fischler, Martin A. and Bolles, Robert C. (1981), “Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography”, *Communications of the ACM*, Vol 24, No 6, pp. 381-395.

Friedman, Jerome, Hastie, Trevor and Tibshirani, Rob (2010), “Regularization Paths for Generalized Linear Models via Coordinate Descent”, *Journal of Statistical Software*, Vol 33, No 1, pp. 1-22.

Geurts, Pierre, Ernst, Damien and Wehenkel, Louis (2006), “Extremely randomized trees”, *Machine Learning*, Vol 63, No 1, pp. 3-42.

Han, Jiawei, Cai, Yandong and Cercone, Nick (1992), “Knowledge discovery in databases: An attribute-oriented approach”, *Proceedings of the 18th VLDB Conference*, pp. 547-559.

Hodge, Victoria and Austin, Jim (2004), “A survey of outlier detection methodologies”, *Artificial Intelligence Review*, Vol 22, No 2, pp. 85-126.

Hoerl, Arthur (1962), “Application of Ridge Analysis to Regression Problems”, *Chemical Engineering Progress*, Vol 58, No 3, pp. 54-59.

Ivakhnenko, A. G. (1973), *Cybernetic Predicting Devices,* CCM Information Corporation.

Kasneci, Gjergji and Gottron, Thomas (2016), “LICON: A Linear Weighting Scheme for the Contribution of Input Variables in Deep Artificial Neural Networks”, *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, ACM, pp. 45-54.

Lathuilière, Stéphane, Mesejo, Pablo, Alameda-Pineda, Xavier and Horaud, Radu (2019), “A comprehensive analysis of deep regression”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol 24, No 9, pp. 2065-2081.

Lundberg, Scott M. and Lee, Su-In (2017), “A unified approach to interpreting model predictions”, *Advances in Neural Information Processing Systems*, pp. 4765-4774.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M. et al. (2011), “Scikit-learn: Machine Learning in Python”, *Journal of Machine Learning Research*, Vol 12, pp. 2825-2830.

Quinlan, Ross J. (1986), “Induction of decision trees”, *Machine Learning*, Vol 1, No 1, pp. 81-106.

Quinlan, Ross J. (1983), “Learning efficient classification procedures and their application to chess end games”, *Machine Learning*, Springer, pp. 463-482.

Ribeiro, Marco Tulio, Singh, Sameer and Guestrin, Carlos (2016), “'Why should I trust you?' Explaining the predictions of any classifier”, *22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1135-1144.

Rifkin, Ryan M. and Lippert, Ross A. (2007), *Notes on Regularized Least Squares. Technical Report*, MIT Computer Science and Artificial Intelligence Laboratory.

Rosenblatt, F. (1958), “The Perceptron: A Probabilistic Model For Information Storage And Organization In The Brain”, *Psychological Review*, Vol 65, No 6, pp. 386-408.

Rumelhart, David E., Hinton, Geoffrey E. and Williams, Ronald J. (1986), “Learning representations by back-propagating errors”, *Nature*, Vol 323, pp. 533-536.

Salakhutdinov, Ruslan, Mnih, Andriy and Hinton, Geoffrey (2007), “Restricted Boltzmann machines for collaborative filtering”, *International Conference on Machine Learning*, pp. 791-798.

Schmidhuber, J. (1992), “Learning complex, extended sequences using the principle of history compression”, *Neural Computation*, Vol 4, pp. 234-242.

Tibshirani, Robert (1996), “Regression shrinkage and selection via the lasso”, *Journal of the Royal Statistical Society: Series B (Methodological),* Vol 58, No 1, pp. 267-288.

Tolvi, Jussi (2004), “Genetic algorithms for outlier detection and variable selection in linear regression models”, *Soft Computing*, Vol 8, No 8, pp. 527-533.

Zou, Hui and Hastie, Trevor (2005), “Regularization and variable selection via the elastic net”, *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, Vol 67, No 2, pp. 301-320

Abbreviations

Con	consolidated reporting level
DPM	data point model
EBA	European Banking Authority
EU	European Union
IFRS	International Financial Reporting Standards
ITS	Implementing Technical Standards
Ind	individual reporting level
JST	Joint Supervisory Team
LSI	less significant institution
NCA	national competent authorities
RMSE	root mean squared error
SI	significant institution
SSM	Single Supervisory Mechanism
SUBA	Supervisory Banking data system
XAI	explainable artificial intelligence
XBRL	eXtensible Business Reporting Language

Conversions used in the tables

"-" Data do not exist/data are not applicable.

"." Data are not yet available.

Acknowledgements
The authors would like to thank Francesco Donat, Francesca Benevolo and Juan-Alberto Sánchez for the fruitful discussions and suggestions during the development of the approach presented. We would also like to thank the Editorial Board of the ECB Statistics Paper Series for their feedback and suggestions.

Stefania Romano
European Central Bank, Frankfurt am Main, Germany; email: Stefania.Romano@ecb.europa.eu

Jose Martinez-Heras
Solenix GmbH, Darmstadt, Germany; email: jose.martinez@solenix.ch

Francesco Natalini Raponi
European Central Bank, Frankfurt am Main, Germany; email: Francesco.Natalini_Raponi@ecb.europa.eu

Gregorio Guidi
European Central Bank, Frankfurt am Main, Germany; email: Gregorio.Guidi@ecb.europa.eu

Thomas Gottron
European Central Bank, Frankfurt am Main, Germany; email: Thomas.Gottron@ecb.europa.eu

Postal address 60640 Frankfurt am Main, Germany
Telephone +49 69 1344 0
Website www.ecb.europa.eu

PDF ISBN 978-92-899-4700-8, ISSN 2314-9248, doi:10.2866/19338, QB-BF-21-002-EN-N