BANCA D'ITALIA

EUROSISTEMA

Questioni di Economia e Finanza

(Occasional Papers)

Application of classification algorithms for the assessment of confirmation to quality remarks

by Fabio Zambuto, Simona Arcuti, Roberto Sabatini and Daniele Zambuto

July 2021

Number 631

BANCA D'ITALIA
EUROSISTEMA

Questioni di Economia e Finanza

(Occasional Papers)

Application of classification algorithms for the assessment of confirmation to quality remarks

by Fabio Zambuto, Simona Arcuti, Roberto Sabatini and Daniele Zambuto

Number 631 – July 2021

The series Occasional Papers *presents studies and documents on issues pertaining to the institutional tasks of the Bank of Italy and the Eurosystem. The* Occasional Papers *appear alongside the* Working Papers *series which are specifically aimed at providing original contributions to economic research.*

The Occasional Papers *include studies conducted within the Bank of Italy, sometimes in cooperation with the Eurosystem or other institutions. The views expressed in the studies are those of the authors and do not involve the responsibility of the institutions to which they belong.*

The series is available online at www.bancaditalia.it .

ISSN 1972-6627 (print)
ISSN 1972-6643 (online)

Printed by the Printing and Publishing Division of the Bank of Italy

APPLICATION OF CLASSIFICATION ALGORITHMS FOR THE ASSESSMENT OF CONFIRMATIONS TO QUALITY REMARKS

by Fabio Zambuto[*], Simona Arcuti[*], Roberto Sabatini[*] and Daniele Zambuto[†]

Abstract

In the context of the data quality management of supervisory banking data, the Bank of Italy receives a significant number of data reports at various intervals from Italian banks. If any anomalies are found, a quality remark is sent back, questioning the data submitted. This process can lead to the bank in question confirming or revising the data it previously transmitted. We propose an innovative methodology, based on text mining and machine learning techniques, for the automatic processing of the data confirmations received from banks. A classification model is employed to predict whether these confirmations should be accepted or rejected based on the reasons provided by the reporting banks, the characteristics of the validation quality checks, and reporting behaviour across the banking system. The model was trained on past cases already labelled by data managers and its performance was assessed against a set of cross-checked cases that were used as gold standard. The empirical findings show that the methodology predicts the correct decisions on recurrent data confirmations and that the performance of the proposed model is comparable to that of data managers currently engaged in data analysis.

JEL Classification: C18, C81, G21.
Keywords: supervisory banking data, data quality management, machine learning, text mining, latent dirichlet allocation, gradient boosting.
DOI: 10.32057/0.QEF.2021/0631

Contents

[*] Bank of Italy, Statistical Data Collection and Processing Directorate.
[†] Bank of Italy, Information Technology Development Directorate.

1. Introduction and motivation[1]

Central banks (CBs) are in charge of the production of high quality statistics based on reliable and timely data reported by financial institutions. Over the last few decades, data collections have experienced a rapid and unprecedented surge in the volume, granularity and frequency of data that reporting agents (RAs) are required to transmit to the authorities. In turn, this has made the activity of ensuring high data quality standards progressively more challenging and time consuming for statisticians working at a CB. As a result, CBs have started to investigate innovative approaches to make their statistical production processes more accurate and efficient (Chakraborty and Joseph, 2017; Bank of International Settlements, 2019).

Recent research in the field of applied statistical analysis has shown that non-traditional techniques based on machine learning and artificial intelligence can offer concrete advantages to CBs in terms of both improving the quality of their statistics and the efficiency of the related processes to compute them, in particular in the new context of increasingly granular surveys. These techniques capture any complex relationships existing in the data reported by financial intermediaries that can be exploited for the prompt detection of potential outliers. Prior studies in this field have applied both supervised and unsupervised algorithms to detect potential anomalies in various types of datasets including securities holdings (Cagala, 2017), balance sheet items (Cusano et al., 2021), and payment services (Zambuto et al., 2020).

The identification of potential anomalies represents only the first step of the data quality management (DQM) process. Generally speaking, outliers correspond to observations that differ "significantly", from a statistical point of view, from the expected data points; however, not all of them necessarily correspond to actual reporting errors, to the extent that some might be due to specific economic or methodological factors ("false positives"). This is why CB data managers submit the potential outliers they have detected to RAs; for each anomaly, the RA can either revise the information previously sent or confirm the data reported. In the latter case, it must add the motivation, which is then analysed by data managers and can be either rejected (a resubmission of the data by the RA is expected) or accepted (the remark is cancelled).

Analysing the motivations received on quality remarks is, thus, a critical step in ensuring a high quality standard and improving the overall DQM system. This process can be highly time-consuming and subject to various inefficiencies, in particular in the presence of more complex surveys. First, the increasing granularity of surveys and the proliferation of validation checks are making the interactions with RAs very complex because the number of potential outliers can be quite large. Second, the cases of "false positives" can become recurrent in the system when the validation checks rely on assumptions that are not valid for all plausible reporting patterns; in turn, a reporting exception, although already known, can affect the data submitted by

[1] We are grateful to Professors Gianluca Cubadda and Alessio Farcomeni (University of Tor Vergata, Rome) for useful comments and fruitful discussions on a preliminary draft of the paper. The views expressed herein are those of the authors and do not necessarily reflect those of the Bank of Italy.

several RAs. Third, the process inevitably requires some degree of judgment by data managers and, then, it is prone to errors that can affect data quality (actual reporting errors are flagged as "normal" data points) or impose an unnecessary burden on RAs (correct data are flagged as outliers). Additionally, the presence of some degree of judgment in the decisions taken by data managers can lead to heterogeneous patterns if similar cases are treated differently over time and across data collections and reporting agents.

In order to address these issues, this study explores the application of machine learning techniques for the automatic processing of confirmations of quality remarks received from RAs. Previous papers in this statistical literature have shown the importance of implementing innovative approaches to identify outliers, while keeping the number of false positives as low as possible in order to mitigate the costs associated with DQM (see, for instance, Zambuto et al., 2020). This work contributes to the stream of the empirical literature by showing that machine learning techniques can also be applied in the DQM process *after* the notification of quality remarks to RAs. Specifically, we propose a classification model to replicate the decision-making process of data managers regarding the acceptance (in the case of false positives) or rejection (in the case of true positives) of confirmations based on the textual explanation provided by RAs, the characteristics of validation checks and the overall reporting behaviour in the system.

This new approach contributes to the improvements of DQM operations in three ways. First, it reduces the need for human intervention in the process by allowing the automatic processing of confirmations corresponding to recurrent cases of true and false positives in the DQM system. Second, by generalizing past decision-making by data managers, the method reduces subjective judgments and ensures a more consistent treatment of similar cases over time and across data collections. Third, since data managers do not have to evaluate an excessive number of confirmations, they can concentrate on new cases that require extensive analyses and identify new exceptions or fallacies in the system.

The empirical analysis focuses on the Single Supervisory Mechanism (SSM) data that the Bank of Italy collects from Italian banks[2]. Our dataset includes information on the outliers confirmed by RAs and regularly stored in the context of the DQM process. This dataset is characterized by a large number of recurrent outliers that are confirmed by RAs and for which the conditions to accept or reject a confirmation have been clarified by the European Banking Authority (EBA). The algorithm is trained on past cases already labelled by data managers as "accepted" or "rejected" and its performance is assessed on a left-out set for which the "true" label was cross-checked and used as gold standard. The empirical findings of the analysis show that the algorithm is able to correctly predict the right decision for the confirmations processed with performance levels (in terms of both sensitivity and specificity) that are comparable, or sometimes even slightly superior, to those of data managers. These results confirm that machine learning techniques can be adopted to improve the overall efficiency of DQM systems and provide advantages that extend beyond the mere identification of potential outliers.

[2] The SSM is the harmonised framework for banking supervision in the EU and comprises the ECB and the participating member states.

The reminder of the paper is organized as follows. In Section 2 the current process for the management of quality remarks and the classification of confirmations is presented. Section 3 describes the new approach based on machine learning models to automatically classify the confirmations of quality remarks. Section 4 shows the empirical results of our analyses. Section 5 gives some concluding remarks regarding the implementation aspects and highlights directions for future research.

2. The Bank of Italy's process to ensure data quality in supervisory reporting: quality remarks and data confirmations

The Bank of Italy regularly collects supervisory data from RAs under EU Commission Implementing Regulations (ITS), laying down implementing technical standards drafted by the EBA. Data collection relies on the technical documentation published by the EBA, which comprises the description of data according to the Data Point Model (DPM) and the related XBRL taxonomy. The Bank of Italy collects information through a number of "surveys", each corresponding to an XBRL module.

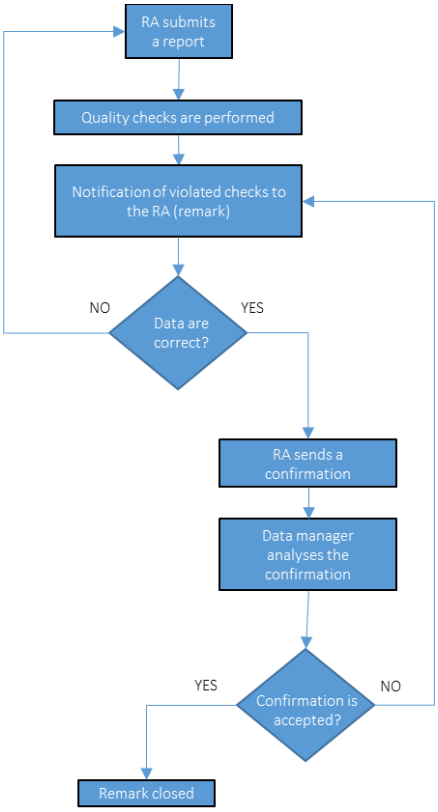
In order to validate the data collected from RAs, the current data quality management is based on a highly automated two-step process. First, the data are validated via a set of quality checks that are carried out automatically upon receipt of the reports. Second, the identified anomalies are communicated to RAs via automatically generated "remark messages". At the end of the validation process, the data are released to the DWH and made available to internal users and to the ECB.

Figure 1 shows the DQM workflow process executed on each reported file. Quality checks of supervisory data essentially comprise formal and deterministic checks. The former verify whether technical standards laid out in a document published by the EBA (the "XBRL filing rules") are fulfilled or not, in which case the reported file is rejected. Deterministic checks are defined in order to verify data correctness and consistency under various aspects such as equivalence or inequality between two aggregates, correct sign of a data point, admitted value of a domain, existence of a data point, and so on. The near totality of deterministic checks consists of validation rules defined by both the EBA and the ECB in cooperation with the national competent authorities (NCBAs). The violation of a deterministic check is notified to the RA by a specific message ("remark"), but it does not imply the rejection of the file.

When a RA receives a remark message related to a deterministic check, it has to verify whether the data indicated in the remark are correct and can react either by *correcting* the data and resubmit the full report, or by *confirming* that the reported data are correct. In this case, the RA has to send a structured message ("confirmation") including the remark reference and the related explanationexplanation, i.e. a free text comment that explains why data are correct. Each individual confirmation is analysed by the Bank of Italy's data manager, who can accept or refuse it. If the confirmation is accepted the remark is considered closed, otherwise a resubmission of the data by the RA is expected. The analysis takes into account the data reported by the RA and the available information about the validation rule and the conditions that make it not applicable.

In practice, most confirmations refer to recurring issues, for which the conditions to admit or reject a confirmation are known. In particular, some issues have already been clarified in the EBA Q&A process together with the conditions under which the violation of a validation check can be regarded as "acceptable". Moreover, it is possible that either the RA that receives the remark or other RAs have already sent confirmations for previous reference dates with a similar explanation.

Figure 1. DQM workflow executed on each reported file.



In the current approach, the analysis of the flow of confirmations in the DQM system is based on a manual process where the data manager has to examine confirmations one by one in order to assess whether they refer to known issues. This activity is quite demanding, in terms of resources and time; moreover, the decision process is based on subjective judgments with the risk of having heterogeneous treatments of similar cases and different reporting burdens on RAs. Any mistake in the outcome of this process has a negative impact on data quality and on the efficiency of the interactions with RAs with a consequent reputational risk.. This is why it is important to increase the level of automation of the DQM process by resorting to advanced statistical techniques that carry out the same steps of the current decision-making process.

The approach we propose in this paper aims at exploiting all the available information in order to classify recurrent cases of confirmations as "to be accepted" or "to be refused" by the data manager. In this way, the data manager will save time in the treatment of known issues and can focus on those for which an in-depth

analysis must be carried out[3]. In this respect, it is important to emphasize from the outset that the approach is data-driven, that is it is based on algorithms that learn how to replicate the decision-making process based on past examples of labelled confirmations. As such, the methodology can be effective in predicting new labels only for those "unseen" confirmations whose characteristics are identical to those of the learning examples. From an operational perspective, this implies that the automated approach we propose is *complementary* to the analyses carried out by data managers that can focus on the identification and analysis of cases of confirmations corresponding to new issues. Periodically, information on such cases can be integrated into the automated procedure by enlarging the set of learning examples used to define the algorithm.

3. A machine learning approach for the classification of confirmations to quality remarks

3.1 Data

In order to perform our analyses we exploit the information related to SSM supervisory reporting data contained in the DQM system of the Bank of Italy. The data falling within the scope of our analysis include information on all the confirmations to remarks sent by the Bank of Italy to RAs for the reference periods between 2018-Q3 to 2019-Q4[4]. This information set includes the various aspects that data managers consider in assessing confirmations (such as, when the confirmation was sent by reporting agents and the content of the text they included).

The unit of analysis in our dataset is the individual confirmation corresponding to a specific remark sent to a given RA, for a given survey and reference period. For each confirmation, the information on the final assessment made by the data manager (i.e. whether it should be accepted or rejected) represent our target outcome[5]. We also extract the textual description provided by the RA illustrating the reasons for violating the check and build a corpus of confirmed remarks. In order to exploit such information into our analyses, we conduct a preliminary manipulation of the textual data through the following series of natural language processing steps:

- conversion to lower case;
- punctuation and stop word removal;
- correction of typos based on ad hoc vocabulary of most common typing errors;
- lemmatisation of words.

[3] New issues refers to validation checks that have not been challenged in the Q&A process. As such, they can be identified with a deterministic rule.

[4] Until 2016 ITS data were collected by Bank of Italy according to a proprietary format and data model. Starting from XXX reference period onward the same data began to be collected according the EBA DPM and the related XBRL technical format. Our study focuses on this latter period in order to avoid structural breakdowns in the reporting schemes.

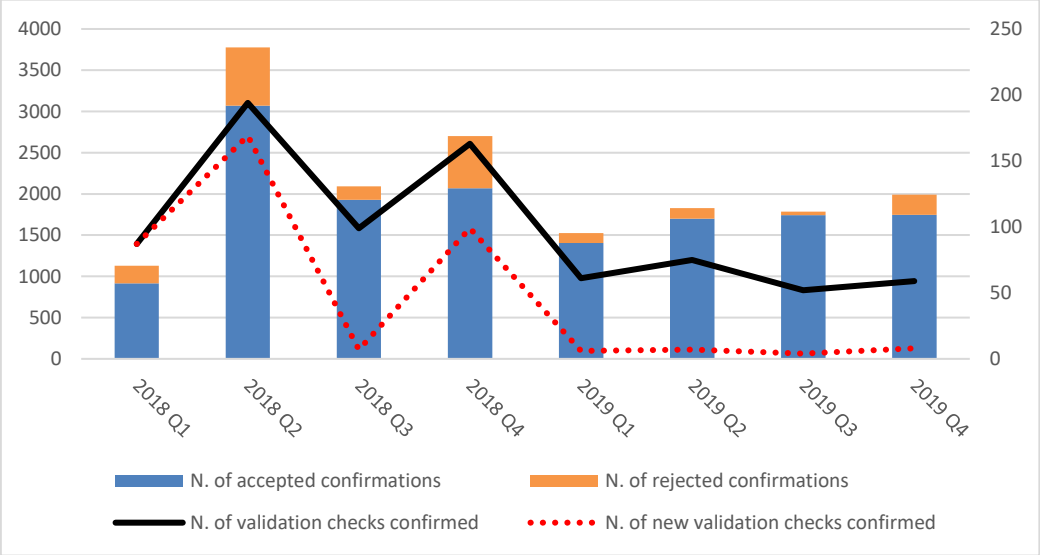
[5] The original label was revised by two additional reviewers in order to detect systemic errors in the dataset. Further details on this data quality management procedure are provided in Section 4.

In addition to textual data, we collect the following information on the DQM workflow executed on each confirmed remark: the time the remark was sent to the RA; the number of remark reminders sent; the time the RA sent the confirmation to the Bank of Italy.

Data on the individual confirmations were complemented with information on the general features of the checks violated, namely the ID of the check, its level of severity and the time of first introduction into the system. Finally, for all the checks included in our dataset, we extract workflow information on all the remaining remarks sent to RAs (i.e. those that were not confirmed).

Overall, the final dataset comprises 13.556 confirmations related to 387 individual checks for which a total of 452.170 remarks were sent to RAs during the period of observation. Figure 2 shows the distribution of the number of confirmations over the considered time span, as well as the number of new validation checks confirmed individual new checks violated in each reference period (i.e. those for which no confirmations have been received in previous quarters) since 2018-Q1. The variation in the number of confirmations across periods mainly reflects the amount of information reported by RAs (and thus of checks performed) and only to a lesser extent the introduction/revision of the system of checks. Indeed, while the number of confirmation tends to remain high during the period of observation, that of newly confirmed checks rapidly stabilizes at very low levels, showing that confirmations tend to be relatively persistent over time.

Figure 2. Number of confirmations, checks and new confirmed checks for the period of observation.



3.2 The prediction problem

The goal of our analysis is to define a classification algorithm able to replicate as accurately as possible the decision-making process for the assessment of confirmations. This problem can be cast in a standard classification setting where the goal is to predict a categorical variable (G) assuming values in C (the set of K possible classes). An optimal decision rule, $\hat{G}$, is sought that minimizes a suitable loss function:

$$\hat{G}(x) = argmin_{g \epsilon C} \sum_{k=1}^{K} L(G_k, g)\, Pr(G_k | X = x) \quad (1)$$

where $\hat{G}(x)$ is a function of a set of observable predictors x and is to be estimated from the data. If L is the 0-1 loss function, the optimal solution to this minimisation problem is the *Bayes classifier*:

$$\hat{G}(x) = max_{g \epsilon C} \Pr(g | X = x) \quad (2)$$

which involves the estimation of the posterior probability of the classes and the assignment to the class corresponding to the largest probability (Hastie et al., 2009). In our context, the target variable *G* can take only two values (*K*=2) describing the decision to take on the confirmation, "accept" or "reject", where the "reject" class is taken as the reference one.

In order to estimate $\hat{G}(x)$ we adopt both traditional and machine learning methods.

Traditional statistical models aim at implementing the Bayes decision rule by directly modelling the probability of the classes. Typically these models rely on rigid structural assumptions on the functional form for the posterior probability and for this reason they offer greater interpretability of model outputs. The statistical models employed in our analysis are logistic regression and penalized logistic regression. In logistic regression the posterior probability of the reference class takes the following form:

$$Pr(G = Reject | X = x) = \frac{\exp(\beta_0 + \beta^T x)}{1 + \exp(\beta_0 + \beta^T x)} \quad (3)$$

Using the logit transformation, the logistic regression model can be re-expressed as a linear model of the log odds (probability of the reference class divided by the probability of the second one):

$$\ln \left[\frac{Pr(G = Reject | X = x)}{Pr(G = Accept | X = x)} \right] = \beta_0 + \beta^T x \quad (4)$$

In this formulation, the parameters of the model are estimated through the maximum-likelihood and the decision rule $\hat{G}(x)$ is described as a linear decision boundary identifying the set of points in the input space for which the posteriors of the two classes are equal. Ultimately, observations are classified depending on the sign of the log odds.

Penalized logistic regression is an extension of the traditional logistic regression model which is aimed at improving predictive performance. The functional form of the model remains the same, but the objective function to optimize is "altered" with the addition of a penalty term whose effect is shrinking the magnitude of the estimated coefficients and improving out-of-sample prediction accuracy (Hastie et al., 2009).

In contrast to statistical models, machine learning (ML) methods do not rely on rigid assumptions on the functional form underlying the model and the parameters of interest are estimated by minimizing the empirical

version of equation (1) (also known as empirical risk). While this comes with a loss of the interpretability of the model, it provides greater flexibility in function estimation as the models can pick up even very complex (non-linear) relationships in the data. In the classification context, this means that models can generate classification rules that partition the input space through non-linear decision boundaries that typically result also in greater out-of-sample prediction accuracy.

In our setting we employ two very popular ML methods: random forest and gradient boosting. Both techniques are based on partitioning algorithms, known as classification and regression trees (CART), that recursively split the input space X into smaller non overlapping regions R_j (called leafs) and then approximate the prediction function $\hat{G}(x)$ in each leaf with a constant γ_j. For classification, the constant is the modal class of observations falling in the leaf. During the growing of the tree each observation is assigned to the majority class in the leaf and a binary splitting criterion – in terms of variable and cut-off point – is determined by minimizing an impurity measure (based on the Gini index or cross-entropy). More formally, the prediction function of a tree can be defines as:

$$\hat{G}(x) = T(x, \theta) = \sum_{j=1}^{J} \gamma_j I(x \in R_j) \tag{5}$$

with $\theta = \{\gamma_j, R_j\}_1^J$. Recursive binary splitting can then be understood as an approximated optimization procedure to minimize the empirical risk (Hastie et al., 2009):

$$\hat{\theta} = \arg\min_\theta \sum_{i=1}^{N} L\big(G_i, T(x_i, \theta)\big) \tag{6}$$

where G_i and x_i indicate, respectively, the observed class and the vector of predictors for the i-th observation in the dataset. Classification trees can fit the data extremely well (i.e. they have low bias) but they also have high variance because their performance is highly sensitive to small changes in the data. The random forest algorithm overcomes the limitations of CARTs by growing an ensemble of trees and combining their predictions. In order to decorrelate the predictions of the tree in the ensemble, every tree is grown on a bootstrapped sample of the data and at each split a random subset of the variables in the input space is considered (Breiman, 2001).

Similarly to the RF algorithm, the gradient boosting model also grows and combines predictions from an ensemble of trees but in a different way. Boosting weak classifiers (i.e. very simple trees) are sequentially applied to repeatedly modified versions of the data and then combined through a weighted majority vote to produce a final prediction at each step (Friedman et al., 2000). The prediction function is thus formalized as follows:

$$\hat{G}_M(x) = \sum_{m=1}^{M} T(x, \theta_m) \tag{7}$$

The sequence of *M* trees are obtained through a forward stagewise additive modelling procedure where the loss function is minimized iteratively by fitting an additive expansion in a set of elementary basis functions to the data:

$$\hat{\theta}_m = \arg\min_{\theta_m} \sum_{i=1}^{N} L(G_i, \hat{G}_{m-1} + T(x_i, \theta_m)) \tag{8}$$

In gradient boosting, the basis functions are obtained through a fast approximation procedure, analogous to steepest descent optimization algorithm, such that at each iteration the term in the expansion is obtained by fitting a tree to the negative gradient of a suitable loss function (for classification the deviance) through OLS regression (Friedman, 2001).

3.3 Model specification

In this section we discuss in more detail the structure of our classification model in terms of the explanatory variables that we include in equation (1) in order to predict the final decision of the data manager on each confirmation.

Within the DQM process of SSM supervisory data the assessment of confirmations is performed on the basis of general operational guidelines set to ensure the treatment of confirmations is as uniform as possible across different surveys. These guidelines can be synthetized as an ordered series of conditions that have to be checked as the analysis proceeds. In the first place, data managers evaluate whether the explanationreasons provided by the RA is formally correct on the basis of relevant references (reporting instructions, usual business operations such as "writedown" or "winding up", EBA Q&As, and so on). If the confirmation is not formally correct it is rejected, otherwise, data managers assess whether the specific case (independently of rejection or acceptance) is consistent with those included in the set of issues that are already known. For confirmations corresponding to known issues a final decision is taken, while in presence of new issues a further, ad hoc analysis is conducted by taking on board not only the consistency of the explanationexplanation provided by the RAs with the reporting regulation, but also contextual factors related to the overall reporting behaviour of RAs (e.g. whether the check was introduced recently, how many RAs violate and confirm it, etc.). Based on the above considerations, our model can be described through the following equation

$$\hat{G}(X) = \hat{G}(X_f, X_s, X_v, X_r) \tag{9}$$

where the decision rule is a function of variables capturing four dimensions: the formal validity of the explanationexplanation provided (X_f), its semantic content (X_s), the characteristics of the validation check (X_v) and the reporting behaviour on the considered check at both individual and system level (X_r).

To measure the formal validity of confirmations, we define various variables describing the general structure of the text provided by the RA. In particular, we include the total number of characters, the number of alphanumeric characters, the number of digits, the total number of words, the ratio between the number of unique words and the total number of words and the number of misspelled words. Moreover, we count the number of occurrences of references to regulations, of competent authorities, of EBA Q&As, of reporting templates and of email exchanges.

In order to capture the semantic content of the explanations provided by RAs we exploit Latent Dirichlet allocation (LDA). LDA specifies a generative probabilistic model of a collection of documents, wherein each document is conceptualized as a random mixture over a set of (latent) topics and each topic represented as a distribution over words (Blei et al., 2003). The ultimate goal of the model is to obtain a low dimensional representation of the collection able to preserve the essential statistical relationships in the data. The representation takes the form of a vector specifying the posterior distribution over the topics of each document and can be subsequently employed in other learning tasks.

To implement LDA in our context we first define a vocabulary of the terms occurring in the corpus of confirmations (including also bigrams) and build a document-term matrix reporting for each confirmation the absolute frequency of each word. Document-word co-occurrences are provided as an input to an LDA model with a pre-specified number of topics (*k*) and for each confirmation a *k*-dimensional representation is obtained. Accordingly, *k* explanatory variables are included in our specification each indicating the probability that a confirmation was generated by the *k*-th topic.

To account for the characteristics of the checks violated we include two categorical variables – indicating the ID of the check and its level of severity, respectively – and control for the number of months since the check was first implemented in the DQM system.

We also control for specific aspects of the DQM workflow in order to capture common reporting patterns on the remarks and the check confirmed. To characterize the reporting behaviour at the remark level for each confirmation we compute the number of days between the reference date and the time the confirmation was sent by the RA and the number of remarks reminders sent to the RAs during this span of time. To capture general reporting patterns at the system level we compute the total number of RAs for which the check was executed at the prior reference date and the corresponding share of RAs that violated the check or that sent confirmations. Similarly, we compute the number of RAs violating the check for the current reference date at the time the confirmation was sent and the corresponding share of RAs that sent confirmations. In addition, we compute the ratio, at the time the confirmation was sent, between the number of RAs violating the check for the current reference and those that have violated it in the testing environment. Finally, we add to our

specification a categorical variable indicating the survey and a trend variable calculated as the number of months since the first reference date in our dataset.

4. Empirical Results

4.1 Estimation and model selection

As explained in the previous section, we estimate $\hat{G}(x)$ through different algorithms: the traditional logit model (LOGIT), the penalized logit model (LOGITp), the random forest (RF) and the gradient boosting (GB). The parameters of the models were fit on a subsample of the data used as *training set*, while left-out samples were used as *test set* to evaluate out of sample performance of the best performing model.

In our estimation procedure two important methodological issues have to be addressed. The first concerns the potential correlation that exists among the observations pertaining to the same RA. Indeed, due to the persistent nature of confirmations in the DQM process, some RAs tend to justify the same issues over time by employing similar lexical forms. This situation could lead to overestimate model performance if very similar observations fall both in the training and the held out sets. As a result, the overall procedure would be likely to overfit the data and result in poor out-of-sample performance on new, unseen data. In order to address this issue the splitting procedure described above was carried out block-wise. Specifically, to obtain the test set we initially sampled 20% of the RAs in our dataset and then included in the final test set all the original observations corresponding to the selected RAs. The same two step approach was adopted to obtain each fold during the cross validation procedure. The final number of observations in the training and test set is reported in Table 1, along with the fraction of rejected confirmation in the two subsamples.

Table 1. Sample observations (training and test set)

	N. of observations	Percentage of rejected confirmations
Training set	10.679	19,8%
Test set	2.877	15,0%
Total sample	13.556	18,8%

The second issue is related to the fact that a relatively small fraction of confirmations in our dataset are rejected by data managers. Such class imbalance may adversely affect the learning process by obscuring any relevant pattern in the data because the algorithm may be pushed towards always predicting the predominant class (Kuhn and Johnson, 2013). In order to mitigate this problem, we ground our model selection procedure on a performance metric that places more emphasis on the minority class, i.e. the F1 score of the rejected class.

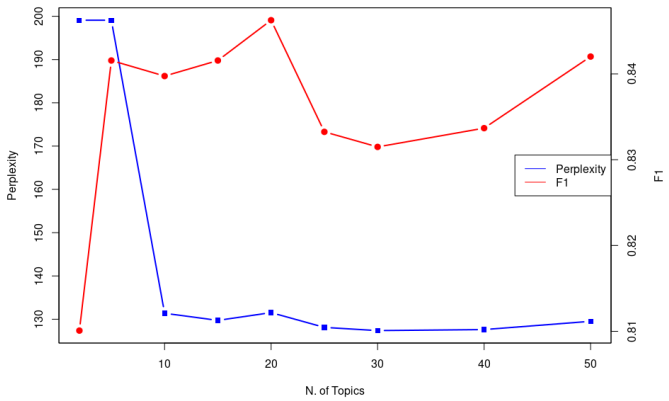
During the estimation phase some of the models also required the calibration of additional *hyperparameters* employed to control the learning process during model fitting. These parameters include the number of variables considered at each split during tree growing for the RF model, and the depth of individual trees and the number of iterations for the GB model. Similarly, the penalization term has to be calibrated for the LOGITp model. Calibration of all these metaparameters was carried out on the training set by employing 10-folds cross validation.

Besides the *hyperparameters* specific to the algorithms employed in order to properly identify our model specifications, our procedure requires to set the number of topics from the LDA model (*k*). Prior research on LDA has suggested different approaches to set the number of topics based on the optimization of specific measures of "fit" for the topics identified. One of the most commonly employed approaches is to set *k* as to minimize the *perplexity* measure, which mathematically is equivalent to the inverse of the geometric mean per-word likelihood (Blei et al., 2003). However, it has been shown that minimizing perplexity is often not correlated with human judgment of the selected topics. Also, in case the topics-based representation of documents is employed as additional features in downstream models, the chosen *k* does not guarantee optimal performance in the ultimate learning task. Thus, since in our context the ultimate goal is classification, we adopt a different approach and choose over a grid of possible numbers of topics based on cross-validation.

The results of the cross validation procedure are summarized in Figures 3 and 4. First, in Figure 3 we show a comparison between the average cross validation F1 score obtained with different number of topics and the perplexity score[6]. For illustrative purposes, the F1-scores are presented only for the GB model, although similar patterns can be observed also for the other models employed. The curve for the F1 score (in red) displays a sharp increase after five topics and reaches a local maximum at twenty topics. A similar pattern is observed for the perplexity score, with the perplexity curve (in blue) showing an elbow at ten topics and then remaining fairly stable for higher values of *k*. Overall, these results confirm that in our context setting *k* to twenty topics should be a reasonable choice.

[6] The score is computed on a held-out test set taken from the overall corpus of confirmation.

Figure 3. Perplexity and average cross-validation F1 scores for different numbers of topics.



Next, we present in Figure 4 a comparison across the different models employed to estimate $\hat{G}(X)$ holding the number of topics fixed. For each model a boxplot is reported describing the distribution of the F1 score computed for the observations iteratively left out during the process. Overall, both the RF model and the GB model show a clear improvement in the performance compared with the more traditional statistical models (LOGIT and LOGITp), while no material differences seem to exist between the latter two. The GB model has the highest performance on average and it is also characterized by a lower dispersion in the F1 score. These results are further investigated in Table 2, which reports additional descriptive statistics on the cross-validated F1 score and the paired t-test on the differences in performance with the best performing model. The estimates confirm that on average the GB model has a higher F1-score and that all the differences with the other models are statistically significant. Based on these measures of performance, the GB model was selected and its out-of-sample performance evaluated in the test set.

Figure 4. Boxplots for the cross-validation F1 score of the four models estimated.

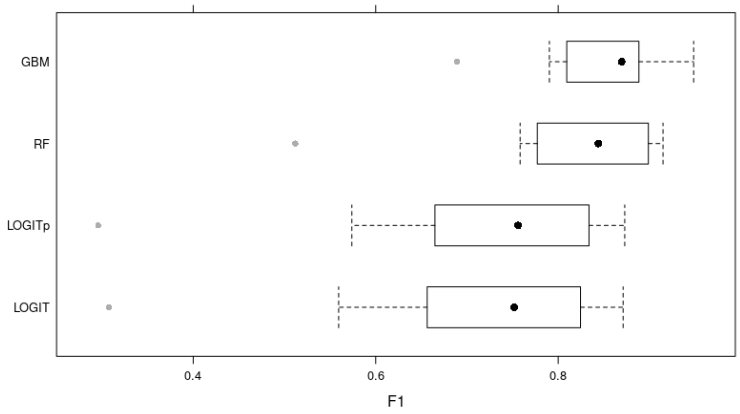


Table 2. Descriptive statistics of the cross-validation F1 and paired t-test for the difference in performance relative to the best performing model.

	Median	Mean	Difference	p-value
LOGIT	0,75	0,71	-0,14	0,01
LOGITp	0,76	0,72	-0,13	0,02
RF	0,84	0,81	-0,03	0,10
GB	0,87	0,85		

4.2 Out-of-sample performance

The performance of the algorithm was evaluated by employing the selected model to predict the final decision on new, "unseen" confirmations in the test set. For benchmark purposes, the performance of the model was compared with the average performance of the data managers on the same set of observations. To estimate the performance of data managers, an iterative data quality management procedure was carried out in order to detect cases of wrong assessments. Specifically, in each iteration, we identified all observations in the test set whose predicted class was different from the original decision taken by the data manager and, for these observations, we asked two additional reviewers to provide a final assessment to be used as ground truth. The procedure was repeated until no more systematic errors were detected. The final results are summarized in Table 3, which reports various performance metrics for two alternative cut-offs levels employed for classification by the model.

Table 3. Performance of the gradient boosting model on the test set.

	(1)	(2)	(3)
	Data manager	GB	
(Cut-off)	-	p=0.5	p=0.10
F1	0,662	0,829	0,821
Sensitivity	0,568	0,726	0,910
Specificity	0,974	0,996	0,946
Precision (positive class)	0,793	0,966	0,748
Precision (negative class)	0,928	0,954	0,983
Accuracy	0,913	0,955	0,941

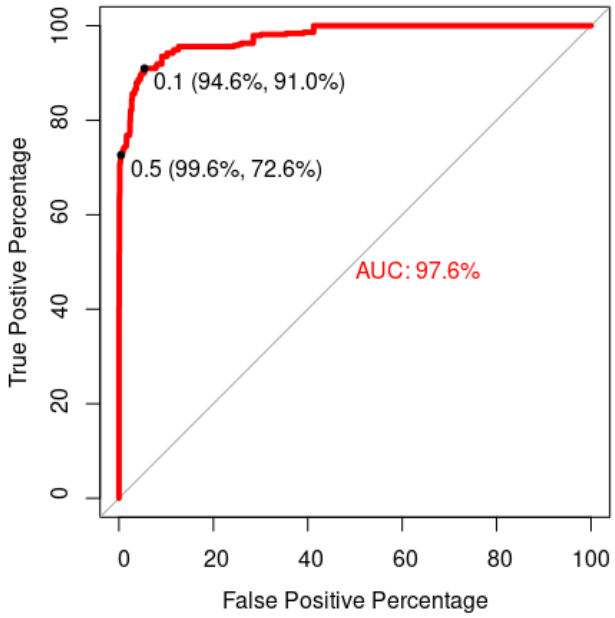
Overall, the model has very good predictive ability, in the sense that its performance is comparable with (sometimes even better than) that of data managers. With a standard cut-off probability of 0.5, the GB model shows a F1 score of 0.82, corresponding to an overall accuracy of 95.5 percent. Both figures are higher than

the corresponding scores for the data managers. A deeper investigation of the predictive performance by classes of the target variable confirms that the algorithm is able to discriminate accurately the minority class. In particular, the specificity (recall) score indicates that more than 72 percent of rejected confirmations are correctly identified as such by the model, while the rejected class is predicted correctly over 96 percent of the times (precision of the positive class). The model does even better in predicting the majority class, as it is able to classify 99.6 % of the accepted confirmations as such while maintaining a precision for the negative class of 95.4 percent.

The performance on the majority class is comparable to those of data managers, while the model seems to do a better job in discriminating the minority class. A possible interpretation of these findings is that the classification rules learned by the algorithm can generalize to some extent "rejection policies" and they are able to "correct" cases where these policies have been applied heterogeneously across different surveys and time periods. Also, the different precision in the classification of the minority class could be explained by the fact that data managers may have a bias towards rejection (i.e. rejecting more often than needed) since false negatives could be perceived as having higher costs (in terms of lowering the quality of data) than false positives (which may eventually be confirmed more than once by RAs).

Considering that false negatives and false positives could be characterized by asymmetric costs in terms of DQM it is also interesting to understand to what extent this information could be incorporated in the classification rule determined by the model. In order to do so we evaluate how the performance of the model changes when the cut-off probability for the rejection of confirmations is set to lower levels. The new cut-off was determined based on the ROC curve of the model (reported in Figure 5) by selecting the level of probability corresponding to the point in the curve closest to the top left corner (representing the perfect classification model). The results of this simulation are presented in column 3 of Table 3 and indicate that the alternative cut-off is associated with only a modest increase in the level of precision for the negative class and more significant reduction in level of precision for the positive class. However, whether the new balance is more preferable or not ultimately depends on the relative costs of false negatives and false positives. For this reason, the selection of an optimal cut-off level is highly dependent on the specific business processes considered (e.g. an optimal threshold for the DQM of SSM supervisory data may not work as well for statistical data) and thus it goes beyond the scope of our analysis. Nonetheless, the simulation exercise shows that such differences could be handled through the algorithm in a flexible way.

Figure 5. ROC curve for the gradient boosting model computed in the test data.



4.3 Robustness checks

In this section we discuss the performance on the test set of alternative models used to automatically classify confirmations. The goal of these additional analyses is to assess the robustness of our procedure with respect to alternative model specifications and estimation techniques.

Model specification is an important aspect to consider because different approaches may be used to capture the semantic content of confirmations through numeric features. To explore this issue we compare the performance of our GB model based on the LDA representation with two alternative approaches. The first is a *Bag of Words* (BoW) model wherein each confirmation is represented by a vector indicating all the terms appearing in the text and their observed frequencies. More specifically, for each term *i* in document *j* the term frequency-inverse document frequency (TF-IDF) measure is reported:

$$TF\_IDF_{ij} = tf_{ij} * log_2\left(\frac{M}{d_i}\right) \tag{10}$$

where tf_{ij} indicates the number of occurrences of term *i* in confirmation *j*, d_i is the number of confirmations including word *i* and *M* is the total number of confirmations in our corpus. Such metric has the advantage of putting greater emphasis on relatively rare words while mitigating the effect of more common terms.

However, while the simplicity of the BoW representation is attractive, it does not take into account word similarity (e.g. synonyms). As a result, documents tend to be closer in the vector space only if they use the same key words. To address this shortcoming the second alternative approach we consider is *Latent Semantic Analysis* (LSA). The central idea of LSA is that words carrying the same or related meanings will often occur

in very similar contexts (Berry et al.). Based on this assumption, the model leverages words co-occurrences within documents to obtain a semantic representation of both words and documents. In this representation two words can be highly related even if they never co-occur together but rather share similar context words. Similarly, documents will be closer in the vector space if they employ very similar words. Mathematically, this is obtained by running the singular value decomposition (SVD) of the term-document matrix (W):

$$W = USV^T \approx U_k S_k V_k^T \tag{11}$$

In which the matrix W is mapped into three components: U and V are the eigenvectors of WW^T and W^TW respectively, and S is a diagonal matrix made by the root of the eigenvalues of WW^T. Typically, a truncated SVD is ultimately used wherein only a portion (k) of the singular values are retained in order to perform a dimensionality reduction that minimizes reconstruction error of the original data. The vectors in US and SV are then employed to represent words and documents respectively[7]. The performance metrics for the GB model based on the LDA, BoW and LSA vector representation are reported in Table 4 (columns 1-3). Both BoW and LSA do not appear to offer significant improvements to model performance. The two approaches are comparable to LDA in terms of specificity and precision for the negative class ("accept"), while they show lower performance in terms of Sensitivity and precision of the positive class ("reject"). In addition, both approaches are less efficient than LDA as they imply highly dimensional vector representations for the documents and ultimately higher computational costs.

While document representation is important to set proper model specifications, model estimation and selection techniques may be instead relevant to deal with class imbalances in the dataset. As explained, an imbalance in the number of examples of rejected and accepted confirmations may prevent our model from learning relevant patterns in the data. In our procedure such imbalance was addressed by choosing an appropriate evaluation metric during the training phase. An alternative and very popular approach is to employ sampling techniques directly aimed at balancing the fraction of positive and negative examples in the training set. In order to explore the robustness of our approach we thus combine our GB model with the following well established sampling procedures: up-sampling, down-sampling and SMOTE. In up-sampling the size of the train set is increased by sampling with replacement cases from the minority class until the two classes have approximately the same size (Ling and Li, 1998). The down-sampling procedure instead contracts the size of the training set by randomly subsampling observations from the majority class until it is reduced to the same size as the minority class (Kuhn and Johnson, 2013). SMOTE (Synthetic Minority Over-sampling Technique), instead, is a data sampling procedure that increases the size of the minority class by synthetizing new observations rather than simply selecting existing observations (Chawla et al., 2002). Specifically, every new data point is created by sampling one observation from the minority class and a generating a random combination of the predictors of

[7] Since there is no general rule to set the number of dimension to retain we adopt the same approach used for the number of topics n LDA and set this parameter through cross-validation.

its nearest neighbours. The three sampling techniques were combined with the gradient boosting model and they were implemented *within* the cross-validation procedure described in section 4, that is, by sampling at each iteration only observations in the training set that were not included in the held out fold. The out of sample performance of the combined estimation procedures was then evaluated on the test set and is summarized in Table 4 (columns 4-6). The results indicate that none of the different combined procedures provides significant improvements in the performance compared to the baseline gradient boosting model with no additional sampling. A possible interpretation of this evidence is that the standard gradient boosting procedure, by iteratively focusing on observations that were incorrectly classified in the previous steps, automatically places more weight on examples of the minority class during model fitting.

Table 4. Performance on the test data of the gradient boosting model combined with different sampling techniques.

	(1)	(2)	(3)	(4)	(5)	(6)
	LDA	Bag of Words	LSA	Upsampling	SMOTE	Downsampling
F1	0,829	0,786	0,795	0,848	0,823	0,837
Sensitivity	0,726	0,682	0,696	0,796	0,740	0,821
Specificity	0,996	0,991	0,990	0,986	0,990	0,975
Pos.Pred.Value	0,966	0,927	0,926	0,907	0,927	0,853
Neg.Pred.Value	0,954	0,946	0,949	0,965	0,956	0,969
Accuracy	0,955	0,944	0,946	0,957	0,952	0,952

5. Conclusions

Previous research has shown that machine learning techniques can contribute to the improvement of the quality of statistics produced by a central bank. These studies have focused on the application of more sophisticated algorithms to identify observations that correspond to potential reporting errors in the data collected from RAs.

This paper contributes to this literature by exploring the advantages that a machine learning approach can offer in the next stage of the DQM process, i.e. once the potential outliers have been detected and communicated to the RAs. Specifically, an innovative methodology is proposed to process the data confirmations received from RAs in response to the Bank of Italy's remarks on quality. Text mining and supervised learning techniques are combined to build a model that is able to replicate the decision-making process of data managers in order to automatically classify these confirmations based on the textual explanations provided by the RAs, the characteristics of validation checks, and reporting behaviour across the banking system. The methodology is applied to the analysis of the confirmations made in response to the quality remarks in relation to SSM supervisory data collected by the Bank of Italy from Italian banks. The model is trained on past cases of confirmations already labelled by data managers and it is tested on a set of confirmations that are cross-checked during the analyses and used as gold standard.

The results show that the model correctly predicts the cases of data confirmations that should be accepted or rejected by data managers, with an accuracy comparable (and sometimes even higher) to that of data managers. The approach offers two fundamental advantages for the DQM process overall.

Firstly, it makes the process more efficient by automatically classifying recurrent cases of confirmations that correspond to true and false positives generated by the DQM system. The automation of the process lowers the costs of interaction between data experts and RAs and ensures a uniform reporting burden across RAs. Secondly, by reducing the number of confirmations that have to be analysed directly, it leaves more room for data managers to focus on the analysis of new cases and identify potential fallacies in data quality checks. This improved ability to process and integrate the feedback loop from RAs is fundamental to continuously improving the DQM system and it is of great importance to statisticians as data becomes more and more granular and new checks are introduced.

The complexity of DQM processes within central banks together with the variety of methodologies available in the machine learning literature, offers several opportunities to extend the present work in various directions.

From an operational point of view, while the focus of our analysis is on a very specific type of data, the approach is quite general and can be adapted to other types of data collection with similar characteristics to those we have considered.

Another important aspect to consider is the typology of confirmations that can be processed automatically by the model. The algorithms employed learn from past examples of decisions made by data managers and so they can provide correct labels only for confirmations related to known issues. Over time, some of these issues

may be fixed by amendments to reporting requirements and new exceptions in the DQM system may emerge. For this reason, it is important to complement the automatic management of confirmations with a full monitoring of the list of active issues in order to filter the type of cases that can be handled by the model.

From a methodological perspective, the approach taken in this paper is a supervised one and is aimed at predicting the decision that a data manager would take in each case of confirmation (accept or reject). A natural extension could be to employ machine learning algorithms for the analysis of confirmations corresponding to new issues, not encountered previously. For example, recurring patterns in past cases of false positives confirmed by RAs could be exploited to estimate the probability that a new case represents a false positive. Similarly, a completely unsupervised approach could be taken to cluster confirmations to group together those containing similar explanations. All this additional information could be integrated into the DQM process to help prioritize the work of data managers and guide their analyses to detect any new fallacies in the DQM system promptly.

References

Bank for International Settlements (2019), "The use of big data analytics and artificial intelligence in central banking", IFC Bulletin, No. 50.

Blei D., Ng A., Jordan M. (2003), "Latent Dirichlet Allocation", *Journal of Machine Learning*, 3, pp. 993-1022.

Breiman L. (2001), "Random forests", *Machine Learning*, 45, pp. 5-32.

Cagala T. (2017), "Improving Data Quality and Closing Data Gaps with Machine Learning", IFC Bulletin, No. 46.

Chakraborty C. and Joseph A. (2017), "Machine learning at central banks", Bank of England Staff Working Paper, No. 674.

Chawla N., Bowyer K., Hall L., Kegelmeyer W. (2002), "SMOTE: Synthetic Minority Over–Sampling Technique", *Journal of Artificial Intelligence Research*, 16(1), pp. 321-357.

Cusano F., Marinelli G., Piermattei S. (2021), "Learning from revisions: a tool for detecting potential errors in banks' balance sheet statistical reporting", Banca d'Italia, Working paper.

Berry M. V., Dumais S.T., O'Brien G. W. (1995), "Using Linear Algebra for Intelligent Information Retrieval", *SIAM Review*, 37, 573-595.

Friedman J. (2001), "Greedy function approximation: A gradient boosting machine", *Annals of Statistics*, 29 (5), pp. 1189-1232.

Friedman J., Hastie T., Tibshirani R. (2000), "Additive logistic regression: a statistical view of boosting", *Annals of Statistics*, 28, pp. 337-307.

Hastie T., Tibshirani R. and Friedman J. (2009), "The Elements of Statistical Learning". Springer.

Kuhn M., Johnson K. (2013), "Applied Predictive Modeling". Springer.

Ling C., Li C. (1998). "Data Mining for Direct Marketing: Problems and solutions", In *Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining*, pp. 73-79.

Zambuto F., Buzzi M. R., Costanzo G., Di Lucido M., La Ganga B., Maddaloni P., Papale F. and Svezia E. (2020), "Quality checks on granular banking data: an experimental approach based on machine learning", Banca d'Italia, Occasional Papers, No. 547.