BANCA D'ITALIA

EUROSISTEMA

Questioni di Economia e Finanza

(Occasional Papers)

A robust record linkage approach for anomaly detection in granular insurance asset reporting

by Vittoria La Serra and Emiliano Svezia

December 2023

Number 821

BANCA D'ITALIA
EUROSISTEMA

Questioni di Economia e Finanza

(Occasional Papers)

A robust record linkage approach for anomaly detection in granular insurance asset reporting

by Vittoria La Serra and Emiliano Svezia

Number 821 – December 2023

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 1972-6643 (online)

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

A ROBUST RECORD LINKAGE APPROACH FOR ANOMALY DETECTION IN GRANULAR INSURANCE ASSET REPORTING

by Vittoria La Serra[a] and Emiliano Svezia[a]

Abstract

Since 2016, insurance corporations have been reporting granular asset data in Solvency II templates on a quarterly basis. Assets are uniquely identified by codes that must be kept stable and consistent over time; nevertheless, due to reporting errors, unexpected changes in these codes may occur, leading to inconsistencies when compiling insurance statistics. The paper addresses this issue as a statistical matching problem and proposes a supervised classification approach to detect such anomalies. Test results show the potential benefits of machine learning techniques to data quality management processes, specifically of a selected random forest model for supervised binary classification, and the efficiency gains arising from automation.

JEL Classification: C18, C81, G22.
Keywords: insurance data, data quality management, record linkage, statistical matching, machine learning.
DOI: 10.32057/0.QEF.2023.0821

Contents

[a] Bank of Italy, Statistical Data Collection and Processing Directorate.

1. Introduction and motivation[*]

In the process of collecting, processing and disseminating statistics, an effective and efficient data quality management (DQM) is of paramount importance for central banks in order to ensure high data quality and to limit the burden of *ex-post* verification on reporters. The automation and precision of DQM processes are becoming increasingly important as databases become larger and more granular.

In the statistical literature, machine learning models are emerging as important tools to approach DQM on very granular data in an automated way, since they generally outperform traditional modelling approaches in prediction tasks (Chakraborty et al., 2017). Focusing on central bank statistics, the Bank of Italy has already successfully applied several machine learning methods to specific DQM processes (see Buzzi et al., 2020, Cusano et al., 2021, Zambuto et al., 2021, Maddaloni et al., 2022) and further research in this field is ongoing.

This paper proposes a machine learning approach in a statistical matching framework to solve - in an accurate and efficient automated way - a DQM issue on insurance granular asset data and to check for anomalies in identification code (ID) reporting.

Specifically, insurance asset IDs are expected to remain unique and consistent over time, i.e. the IDs assigned by insurance corporations (ICs) are not expected to change throughout the reporting history of the assets. However, unexpected changes in the ID for the same asset may occur between two consecutive reporting dates. This is either due to updates to the requirements or, more commonly, to reporting errors. Either way, such changes have major implications for the work of supervisory authorities and central banks, since they may be misinterpreted as a signal that a previously reported asset was removed from the IC's portfolio and that a new asset was added to the portfolio when, in fact, this is not the case. This reporting behaviour raises DQM issues when analysing the time series of assets and compiling the IC statistics that are then disseminated.

This work stems from an ESCB joint project within the "network of experts on machine learning", established by the ECB's Statistical Committee, and is an extension of a previous paper by the same authors (La Serra, Svezia, 2022)[2]: the main innovations of this paper are the assessment of the temporal robustness of the proposed methodology, a deeper performance analysis for different asset types, and a first validation with the reporting agents during the production rounds.

The paper is structured as follows. Section 2 describes the data from which the dataset used in the analysis is derived, presenting its structure and details on the Italian case. Section 3 proposes a record linkage approach based on machine learning models for classification; it assesses different models for the Italian dataset, selects a robust and high-performance random forest, and presents the results. Section 4 illustrates the test results for

[*] The views expressed herein are those of the authors and do not necessarily reflect those of the Bank of Italy.

[2] This paper benefits from the suggestions and remarks received during the presentation of the previous work by the same authors, held at the 11th biennal conference of the Irving Fisher Committee of the Bank for International Settlements (Basel, 2022), where it was awarded as the 'best paper by a young statistician'.

all types of assets and for specific asset types. Finally, the main conclusions summarize the advantages of the proposed approach and open up new avenues for future research.

2. Data description

Since 2016 European insurance corporations (ICs) report to their national supervisory authorities[3] and the national central banks quarterly data on their individual balance sheets. The data is organised in templates according to the Solvency II Directive[4]. They provide very granular and highly valuable information especially with template S.06.02 which contains asset-by-asset information on the single holdings of insurance corporations, showing the investments in debt securities, equity and investment fund shares, as well as loans, deposits and properties.

Template S.06.02 allows, on the one hand, supervisory authorities to perform a comprehensive and detailed risk assessment on insurance undertakings and, on the other, central banks to compile statistics about the insurance sector, useful to analyse its interconnections within the financial system and to gather knowledge on households wealth and income from insurance policies. The supervisory template is enriched with specific information for the statistical purposes.

More in detail, template S.06.02 comprises quantitative information on each position held, such as the market and nominal value, quantity and accrued interest of the asset, along with qualitative features, which include – wherever applicable – the type of insurance undertaking, the type of asset, the issuer and/or counterparty sector, the issuer and/or counterparty area, the currency, the issue and maturity dates, the name of the issuer, the description of the asset.

Each asset in the template is reported with an identification code (ID). Assets' IDs are standardized in most cases (e.g. ISIN codes for securities), although in some cases insurance corporations can report their assets with internally assigned codes (CAU, Code Attributed by the Undertaking).

The dataset used in the paper consists of the S.06.02 template reported by Italian ICs between 2019 and 2022 and it is also integrated with attributes originated from the Centralised Securities Database (CSDB) - the ESCB harmonised security registry. In detail, the population of Italian ICs is composed of around 100 entities; overall, reported data comprise almost 30 reporting quarters and around 70,000 assets at each period. Such data is collected by IVASS, which releases it to Banca d'Italia in order to compile ESCB statistics on insurance sector.

On average, in each quarter, there is a turnover of 8% in number and 4% in market value share for the reported assets. In line with the reporting instructions, assets' IDs that are only reported in one of two adjacent quarters should consist in new purchased or sold assets. However, these also include the cases of erroneous changes in

[3] According to the Implementing Technical Standards (ITS) drawn by EIOPA: Commission Implementing Regulation (EU) 2015/2450 of 2 December 2015 and following amendments, laying down implementing technical standards with regard to the templates for the submission of information to the supervisory authorities, according to Directive 2009/138/EC of the European Parliament and of the Council.

[4] Directive 2009/138/EC of the European Parliament and of the Council of 25 November 2009 on the taking-up and pursuit of the business of Insurance and Reinsurance.

the codes, whose exact percentage in the data is unknown. Therefore, 8% can be taken as the maximum expected percentage of cases of anomaly, which highlights how, even having a limited impact on the general data quality, errors in IDs cannot be neglected.

3. The proposed machine learning approach

In this section we illustrate the literature regarding record linkage, the preprocessing of insurance data and the training, validation and testing of the machine learning models.

3.1 Record linkage in literature

Statistical matching techniques, as described in D'Orazio et al. (2006), have the objective to draw information from two (or more) different datasets by linking them with respect to some common observed variables. Such techniques were originally proposed with the aim of data integration, i.e. to link two (or more) datasets coming from independent surveys and build a richer dataset containing information from both (Okner, 1972).

A specific case of statistical matching is record linkage, which is applied when the statistical units in two datasets are supposed to be at least partially overlapping (D'Orazio et al., 2006) and the objective of the analysis is to identify the list of common units between the two.

The topic was first introduced and formalized by Fellegi et al. (1969) and a classical approach was then proposed by Jaro (1989); different methodologies for performing record linkage have later been proposed, such as mixture models (Larsen et al., 2001) and Bayesian approaches (Fortini et al., 2001; Tancredi et al., 2011). More recent contributions to record linkage make use of machine learning techniques (Feigenbaum, 2016, Rijpma et al., 2020).

Since mid-Nineties, many applications of record linkage have concerned the issue of linking historical census data, as in the works of Ferrie (1996), Rosenwaike et al. (1998) and Ruggle (2002); more recent literature on record linkage concerns different real life issues, such as health issues (Mumme et al., 2022; Heidinger et al., 2022), deduplication issues (Christen et al., 2011; Tancredi et al., 2020) and crime and fraud detection (Vatsalan et al., 2013), among others.

The issue of unexpected changes of IDs in insurance data introduced above in Section 1 can be approached as a record linkage problem; the current work uses the more recent machine learning framework.

3.2 Building the comparison matrix

As described in Section 1, each asset in a quarter is identified by a unique ID and reported with a set of qualitative and quantitative features. If a change in an asset's ID occurs, so that an insurance corporation reports

asset "a" in quarter $\boldsymbol{Q_t}$ and recodes it as "b" in quarter $\boldsymbol{Q_{t+1}}$, it is expected for the reported features of the two apparently different assets "a" and "b" to take the same values, since they actually refer to the same asset. Comparing the reported features of the two assets is therefore necessary to assess whether their difference in ID is in fact an anomaly, stemming from an unexpected change that has taken place.

As in a record linkage framework, two datasets of assets, each referring to two adjacent reporting quarters $\{\boldsymbol{Q_t}, \boldsymbol{Q_{t+1}}\}$, can be compared to assess whether there are common units between the two datasets; each unit in a dataset is an asset, identified by its ID and its reporting IC.

Assets in the two quarters are compared with respect to the observed features and such comparisons are carried out using distance measures, one for each feature's type, whether categorical (nominal or ordinal), numerical or textual. Distances are computed in order to build a comparison matrix, as reported in Table 1. Each row in the matrix refers to a pair of assets from the two adjacent quarters and each column refers to an observed feature, either nominal, ordinal, numerical or textual, for which a distance measure $\boldsymbol{d}(.)$ is chosen. The $\boldsymbol{d_f(a,b)}$ distance measure for two assets a and b on a feature f is a value calculated in [0,1], where the endpoints of the interval respectively indicate minimum and maximum distance between the two observed values for feature f.

Nominal variables, such as counterparty sector or issuer area, are compared with an overlap measure (Boriah et al., 2008), taking 0 as a measure of minimum distance if the reported values are equal and 1 otherwise; this measure produces binary distances (0 or 1), therefore it is only used for the comparison of nominal variables, where it is meaningful for couples of values to be only evaluated as equal or not.
Ordinal variables, such as the categorized maturity date, are compared via the Manhattan distance, while numerical variables, as the assets' market value, are compared using a Euclidean[5] distance.
Lastly, textual variables as the assets' description are compared using a Levenshtein measure for strings; given two strings s1 and s2, the Levenshtein distance is defined as the total number of deletions, insertions, or substitutions required to transform s1 into s2 (Haldar et al., 2011).

Each computed distance is normalized to take values in the interval [0,1]. This is done so that distances for different features can be compared and no measure is affected by extreme values in its distribution, either very large or very small.

The final step of the construction of the comparison matrix consists in adding a further column, referred to as the "status" variable. This variable takes value "match" if the two codes are equal and "non-match" otherwise. In the example in Table 1, the couple (A, A) is a match, while (A, B) is a non-match.

[5] In the current work, only one numerical variable is used for comparison. Therefore, the Euclidean distance is equivalent to the Manhattan distance.

The columns in the comparison matrix are used as input (covariates) to supervised statistical models together with the status of each pair that is the binary target variable to be predicted with values "match" or "non-match".

Table 1. Input to supervised models: the comparison matrix and the target variable

			COMPARISON MATRIX			
Asset codes		TARGET VARIABLE	Nominal	Ordinal	Numerical	Textual
Q_t	Q_{t+1}	Status	$i \in \{1 \dots n_i\}$	$j \in \{1 \dots n_j\}$	$k \in \{1 \dots n_k\}$	$w \in \{1 \dots n_w\}$
a	a	Match	$d_i^1(a,a) \dots d_i^{n_i}(a,a)$	$d_j^1(a,a) \dots d_j^{n_j}(a,a)$	$d_k^1(a,a) \dots d_k^{n_k}(a,a)$	$d_w^1(a,a) \dots d_w^{n_w}(a,a)$
a	b	Non-match	$d_i^1(a,b) \dots d_i^{n_i}(a,b)$	$d_j^1(a,b) \dots d_j^{n_j}(a,b)$	$d_k^1(a,b) \dots d_k^{n_k}(a,b)$	$d_w^1(a,b) \dots d_w^{n_w}(a,b)$
b	b	Match	$d_i^1(b,b) \dots d_i^{n_i}(b,b)$	$d_j^1(b,b) \dots d_j^{n_j}(b,b)$	$d_k^1(b,b) \dots d_k^{n_k}(b,b)$	$d_w^1(b,b) \dots d_w^{n_w}(b,b)$
…	…	…	…	…	…	…

From the Italian insurance database, assets from two subsequent reporting quarters can be considered; at each quarter, around 70,000 assets are reported. With the goal of detecting the changes in IDs that ICs have reported between couples of adjacent quarters, comparison is only made for pairs referring to the same ICs. Even with this constraint, the number of rows in the matrix, i.e. the number of compared pairs between two quarters, approaches on average 150 million. Given the size of the dataset, it would be impossible for data analysts to manually check all pairs of assets.

3.2.1 Sampling the comparison matrix: training and test sets

As presented in the Data Description section, in the Italian data there is a turnover of 8% in number for the reported assets, on average, in each quarter. Observing this, 8% can be taken as the maximum expected percentage of cases of anomalies, i.e. potential errors in the reporting codes. Based on prior experience on insurance data, we do not expect the true (and unknown) percentage to exceed such threshold. However, it must be ensured that the proposed methodology can perform outlier detection even with larger or smaller percentages of anomalies in the data. For this reason, we tested the approach for different percentages of unbabalance p ranging from 1% (extreme unbalance) to 50% (perfect balance).

The comparison matrix, built as described above, is afterwards split into a "training set" and a "test set", respectively including 80% and 20% of the data. Moreover, both datasets are stratified with respect to the "asset type" feature, in order to obtain datasets in which each asset type is represented, and sampled to be unbalanced with respect to the target variable to present p% cases of match and $(1-p)$% cases of non-match.

More in detail, training and test datasets are built through the following steps.

For a fixed unbalance proportion p and for each asset type, all couples of assets are considered by filtering that specific asset type in the original comparison matrix; the subset is over/subsampled in order to have p% of

cases of match and $(1-p)\%$ cases of non-match and simultaneously randomly split in 80% (training set) and 20% (test set).

In our application to Italian data, in practice, the described sampling with respect to the unbalance proportion always consists in taking all cases of match and subsampling the cases of non-match, since the former are the minority class in the comparison matrix, while the latter are fictitiously generated through the Cartesian product of all couples of assets.[6]

The steps presented above in this section are repeated for twelve couples of subsequent reporting quarters, starting from 2019Q1-2019Q2 until 2021Q4-2022Q1. Even if data is available since 2016, the reason for the choice of the "starting point" in 2019 is that, based on prior experience from working in the field of Central Bank reporting and statistical production, the quality of reported data gradually increases with time, after the introduction of a new regulation. Also, the most recent data can be more representative of the current asset portfolio.

All the couples of quarters, except the last one, are used in the validation phase for the fine-tuning of the hyperparameters of the models and for selection of the best model, as described below in sub-Section 3.3; on data from the last couple of quarters, test results from the chosen model are presented in detail in Section 4.

3.3 Model selection

3.3.1 Investigated classes of model

For a selected couple of adjacent quarters (Q_t, Q_{t+1}) with t in {2019Q1, … 2021Q4} and a fixed percentage p of unbalance in the target, with p in $\{1\%, \dots 50\%\}$, the training and test datasets from the comparison matrix are built, as described in the previous section.

On each training and test dataset, three classes of supervised classification models are fitted: the logit model, random forests and neural networks.

The logit is used as a benchmark, due to the fact that it is a classical and probabilistic logistic regression model, a high-performing yet easy-to-interpret classifier (Feigenbaum, 2016). The logit is a regression type of model for binary response variables which makes use of the standard logistic function, i.e. a sigmoid function.

[6]The proposed methodology to stratify the comparison matrix also applies to cases in which the match cases represent the majority class.

The random forest is a tree-based ensemble model and its hyperparameters[7] are the number of bootstrap samples and the number of variables to use at each split. The former represents the number of random trees to fit and the latter ensures that trees are not very similar to each other. Defined by Howard et al. (2012) as "the most successful general-purpose algorithm in modern times", random forests are used in a wide range of real life problems, such as ecology issues (Prasad et al., 2006; Cutler et al., 2007), bioinformatics (Diaz-Uriarte et al., 2006) or econometrics (Varian, 2014).

Finally, neural networks are considered. These are made of different layers, the first and last respectively called "input" and "output" layers and the inner ones being called the "hidden" layers.[8] Each hidden layer is composed of "nodes", interconnected in a directed and weighted graph; each node is a regression or classification model. As presented by Bishop in 1995, neural networks are notoriously powerful methods for prediction and are widely used in many areas, such as social science, engineering, economic, business or finance (Adebiyi et al., 2014), image or speech recognition (Egmont-Petersen et al., 2002; Ossama et al., 2014). Its usage is due to its desirable features, such as it being data-driven and self-adaptive with a few prior assumptions (Khashei et al., 2010).

3.3.2 Validation phase

In this section we assess the performance of the three classes of models for different hyperparameters, fitted on all couples of quarters from 2019Q1-2019Q2 until 2021Q3-2021Q4, and we select the best model.

For the logit model, since it is just used as a benchmark, no fine-tuning of its hyperparameter is considered: the probability threshold for classification is fixed to 0.5; the random forest is trained and tested with different numbers for its hyperparameters, the number of trees and number of variables to use at each split (*mtry*[9]), and the neural network is trained and tested with different numbers of nodes in its single hidden layer.

For each couple of quarters, averaging over the results obtained on differently unbalanced data, test results for a model can be summarised in an average ROC (Receiver Operating Characteristics) curve, with its corresponding AUC (Area Under Curve) index. Each curve is built with average false positive and true positive rates and it varies with the probability threshold for classification.

The AUC indexes for all the considered models are reported in Table 2 for all couples of reporting periods; in yellow is highlighted the best performing model for each couple of quarters[10].

[7] Model's hyperparameters and parameters need to be distinguished: the former are selected by the researcher depending on the objectives of the analysis while the latter are estimated in the training phase.
[8] In the current paper, no deep neural networks are used but only networks with one single hidden layer.
[9] Notation from R programming language.
[10] The same (best) performance might be achieved by multiple models in the same couple of quarters.

Table 2. Average AUC for class of model and hyperparameters

	Logit	Neural network			Random forest								
					100 trees			200 trees			300 trees		
	threshold = 0.5	10 nodes	20 nodes	30 nodes	mtry = 5	mtry = 6	mtry = 7	mtry = 5	mtry = 6	mtry = 7	mtry = 5	mtry = 6	mtry = 7
2019Q1-2019Q2	97.69	97.31	97.62	97.69	**98.54**	98.33	98.31	98.48	98.37	98.31	98.44	98.41	98.37
2019Q2-2019Q3	98.02	96.72	97.65	96.71	98.48	98.43	98.39	98.47	98.46	98.41	**98.52**	98.51	98.45
2019Q3-2019Q4	99.51	99.81	**99.93**	99.92	99.91	99.92	99.91	99.92	99.91	99.91	99.92	99.92	99.90
2019Q4-2020Q1	97.77	99.44	99.59	99.57	99.64	99.63	99.62	**99.66**	99.62	99.60	99.66	99.63	99.62
2020Q1-2020Q2	97.70	97.82	97.18	96.98	98.78	98.55	98.58	**98.81**	98.61	98.59	98.76	98.69	98.59
2020Q2-2020Q3	98.00	97.53	98.94	97.63	**98.99**	98.77	98.71	98.90	98.69	98.67	98.92	98.74	98.71
2020Q3-2020Q4	97.84	97.72	**99.15**	97.79	98.87	98.59	98.47	98.76	98.59	98.46	98.71	98.66	98.44
2020Q4-2021Q1	97.77	97.41	97.78	97.54	98.75	98.69	98.54	98.78	98.61	98.54	**98.81**	98.58	98.55
2021Q1-2021Q2	98.12	98.21	97.77	98.03	98.88	98.79	98.69	**98.95**	98.76	98.73	98.87	98.82	98.75
2021Q2-2021Q3	98.12	97.49	98.48	98.19	**99.04**	98.89	98.92	**99.04**	98.89	98.89	99.02	98.91	98.89
2021Q3-2021Q4	97.93	98.08	**99.09**	98.05	98.83	98.78	98.77	98.98	98.81	98.78	98.99	98.75	98.77
Average	98.04	97.96	98.47	98.01	**98.97**	98.85	98.81	**98.98**	98.85	98.81	98.97	98.87	98.82

The results reported in Table 2 show that all three models perform well on the different couples of quarters, since all AUC indexes are very large, never under 96%, and the mean values for the models are all around 98%, the logit presenting the smallest value, the random forest presenting the largest one.

Among the three classes of models, performance for the benchmark one (the logit) are always equal or lower to those of the other models. As it is observable, in most couples of quarters, performance for the random forest model are higher than those of the neural network; it is then interesting to observe how the neural network with 20 neurons in the hidden layer slightly outperforms the random forest only in the Q3-Q4 couples of quarters.

The presented AUC indexes can be used for the fine-tuning of the hyperparameters in the models.

For the neural network, a value of 20 nodes in the hidden layer is selected as the best one, as this is the one that provides the best results in most cases; concerning the random forest, instead, in order to select the best hyperparameter, a consideration is made: there is no need for a very complex model if this does not have a substantial impact of increase on the performance measure. Indeed, a random forest with 100 trees and 5 *mtry* ensures an average AUC of 98.97%, which is just 0.01 smaller than the maximum average AUC observed for 200 trees and same *mtry*.

For completeness, all average ROC curves from all the couples of reporting quarters considered and the best combination of hyperparameters for each class of model selected in the validation phase can be observed in Appendix. The superiority of the random forest model against the others can be clearly observed in the figures in Appendix, where its average ROC curves often show larger true positive rates over false positive rates with respect to the other models.

In light of the presented results, with the goal of selecting a unique model for future applications, the random forest model with the selected hyperparameters (100 trees, 5 *mtry*) is chosen as the best among the investigated ones[11].

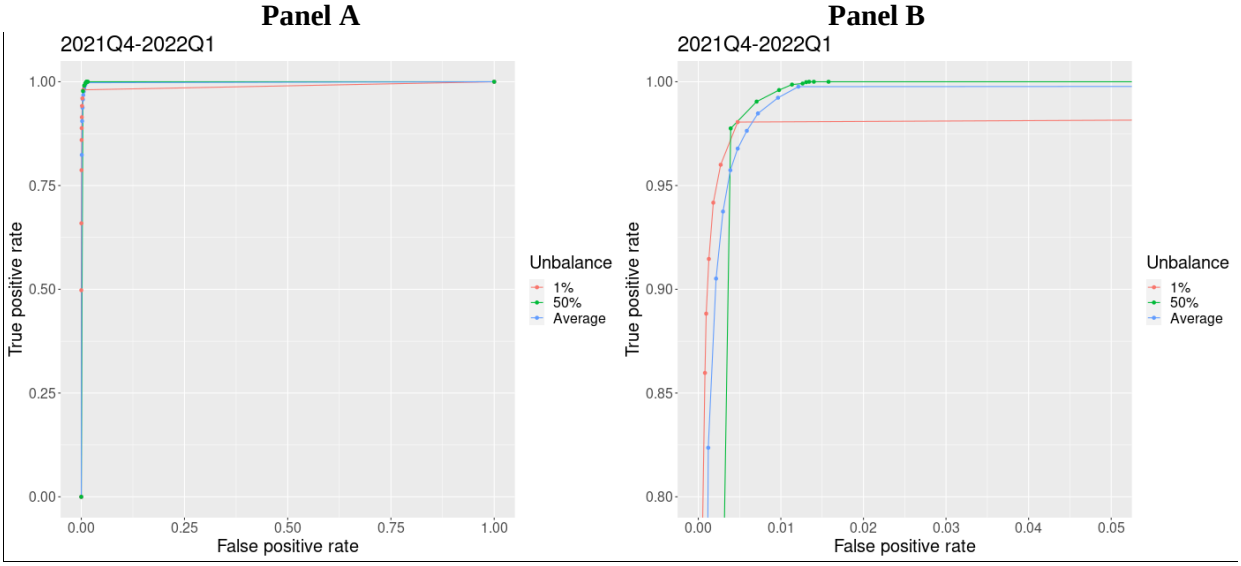
As observable from Table 2, all the investigated models show stable average AUC over the different couples of quarters, proving the robustness of the performance of the proposed approach in time.

4. Test results

In the previous section we have shown the robustness of the approach to different couples of quarters; in this section, with reference to the couple of quarters 2021Q4-2022Q1, test results for the selected random forest are presented in detail to prove also robustness to unbalance in terms of percentage of matches.

In Figure 1, ROC curves for the selected random forest are shown for different percentages of unbalance; the corresponding values for the AUC indexes are observable in Table A.3 in Appendix, showing satisfying results for all unbalances in the dataset, from 1% to 50%. Moreover, as observable in the plots in Figure 1, the average ROC curve is closer to the one for balanced data (50%) than to the one for extremely unbalanced data (1%). As expected, a high true positive rate for a highly unbalance dataset is reached only with low thresholds, while a high true positive rate for a balanced dataset is achieved also with high thresholds.

Figure 1. ROC curves for differently unbalanced test sets



[11] Highlighted in green in Table 2.

Panels A and B both show the ROC curves for the most (1%) and least (50%) unbalanced test sets; panel A shows the whole curves, while panel B focuses on smaller ranges of the axes, to better spot the differences. Each point in the ROC curves refers to a specific probability threshold for classification; from left to right, the threshold varies from 1 to 0.

4.1 Overall results

In the previous sections we have shown the robustness of the chosen model to different couples of quarters and different unbalance percentages in the data. For this reason, in this section, more test results for the selected best random forest are shown for the couple of quarters 2021Q4-2022Q1 and an unbalance percentage $p = 5\%$. This percentage is chosen for illustrative purpose, being smaller than 8%, namely the maximum expected unbalance proportion in the dataset (see Section 1).

Performance measures for the model are presented in Table 3, varying only with the probability threshold for classification. The presented indexes are accuracy, balanced accuracy, true positive rate (TPR), true negative rate (TNR), false discovery rate (FDR) and the difference between true positive rate and false discovery rate[12].

[12] Formulas for the different indexes are detailed in Table A.2 in Appendix, based on the confusion matrix in Table A.1.

Table 3. Model performance indexes for the random forest selected model (p=5%)

(percentage values)

	Probability threshold for classification									
	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	Average
Accuracy	98.82	99.23	99.45	99.53	99.58	99.58	99.54	99.40	99.08	99.36
Balanced accuracy	99.36	99.25	99.05	98.76	98.34	97.66	97.04	94.94	91.45	97.32
TPR	99.95	99.28	98.61	97.89	96.97	95.54	94.26	90.00	82.96	95.05
TNR	98.76	99.23	99.49	99.62	99.72	99.79	99.82	99.89	99.93	99.58
FDR	19.03	12.83	8.90	6.87	5.27	4.04	3.55	2.23	1.58	7.15
TPR-FDR	80.92	86.45	89.72	91.02	91.70	91.49	90.71	87.76	81.39	87.91

True negative rate (TNR) shows very good results, remaining stably around a 99% value for all thresholds. Making 95% of the unbalanced test set, the cases of non-match are naturally easier to be detected by the classification model.

A more interesting result is the ability of the model to correctly identify the minority class in the data, i.e. the matches (5%): this ability is measured by the TPR, which is the percentage of correctly classified cases of match among the true cases of match; this is a quantity that one would like to maximise as much as possible. As observable in the table, the average TPR is indeed very large, above 95%.

General model performance measures such as accuracy and balanced accuracy show that the model correctly identifies true cases of match and true cases of non-match with high frequencies, for all probability thresholds. Mean values for the two measures are respectively 99.36% and 97.32%.

Accuracy is computed as the percentage of correctly classified cases - both positive and negative - among all cases. It should be highlighted that, in an unbalanced test dataset, the minimum acceptable value for this metric is the percentage referred to the majority class: in fact, a trivial model predicting always this class would achieve an accuracy equal to the majority class percentage; in the application in this section, the majority class is represented by the proportion of non-matches cases in the test set (95%). As observable from the first row in the matrix, values for the accuracy are always higher than this minimum.

Despite being simple and intuitive for a model's performance, accuracy is strongly affected by unbalances in the target variable. The topic has been deeply analysed, as in the works of Chawla et al. (2004), Daskalaki et al. (2006) and Elazmeh et al. (2006). In the case of strongly unbalanced data, an alternative general model performance that can be used is balanced accuracy, which is computed as the simple mean between TPR and TNR. Since the current results are referred to a 5-95% unbalanced test dataset, this metric is more appropriate to measure the model's performance. Considering again the trivial model predicting always the majority class (in this case, 95% of non-matches), the minimum acceptable balanced accuracy would be 50%, where TNR and TPR would respectively be 100% and 0%. As observable from Table 3, even for large values of the

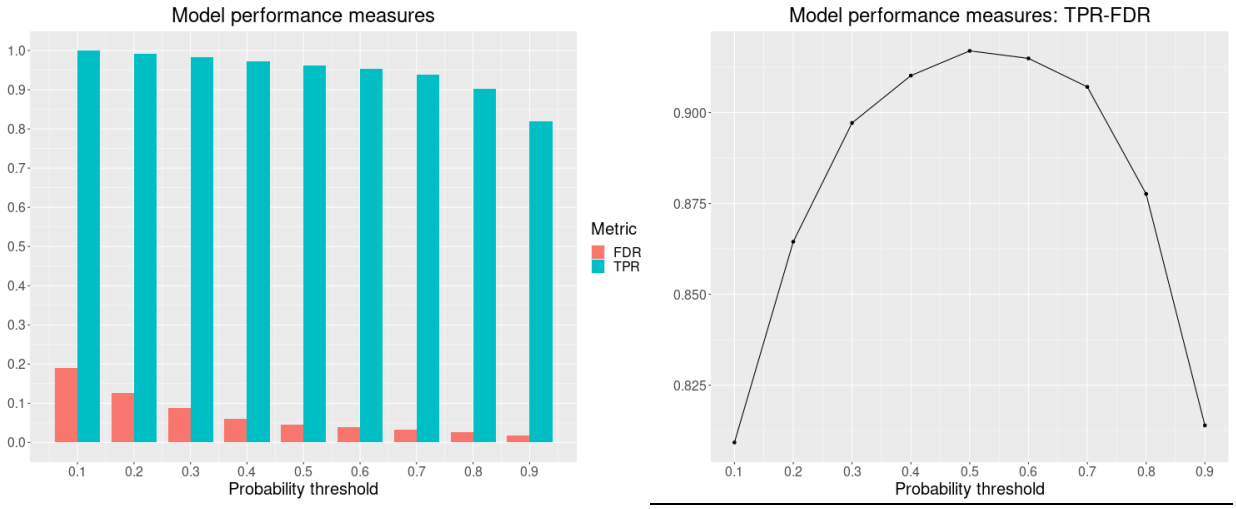
probability threshold, balanced accuracy is lower than accuracy but always much larger than the 50% minimum.

Taking into consideration that results vary with the probability threshold for classification, we also want to choose the threshold which optimizes the model's performance for our work. The selection of the probability threshold is conducted by analyzing specific performance metrics that we consider to be relevant: true positive rate (TPR), false discovery rate (FDR) and the difference between the two. These metrics are presented in Table 3 and in Figure 2.

Benjamini et al. (2001) define FDR as the "expected proportion of false discoveries among the discoveries". Given a binary confusion matrix, FDR is the percentage of negative cases that the model incorrectly classifies as positive cases; in the current analysis, FDR measures the percentage of non-matching assets that are erroneously classified as matches by the model and it is therefore a DQM cost to minimize.

Figure 2. Model performance measures for 2021Q4-2022Q1 (TPR, FDR, TPR-FDR)

Panel A **Panel B**



Panel A shows TPR and FDR while panel B shows the difference between the two indexes for different probability thresholds.

As observable in Figure 2 - panel A, both the true positive rate and false discovery rate slowly decrease with the probability threshold increasing.

As observable in Table 3, for instance, taking the lowest threshold for classification, the selected model ensures a 99.95% rate of correctly classified cases of match, with the consequence of a 19.03% incorrectly classified

cases of non-match; instead, taking the highest threshold, less than 2% false discoveries are made but only 82.96% cases of match are correctly identified.

Therefore, a trade-off between the two indexes must be found in order to choose an appropriate probability threshold for the model. However, the two rates do not have the same weight in the current analysis: although false discovery rate is a DQM cost to minimize, maximizing the true positive rate is considered as a priority for the model's effectiveness. Detecting most of the true cases of match is in fact the goal of the analysis and it is therefore desirable to select a lower probability threshold for classification which would ensure to reach the goal, even if that implies that some cases of non-match are erroneously classified as matches.

To assess for the best threshold, the difference between TPR and FDR is calculated and reported in Table 3 and Figure 2 - panel B. The maximum value for the index is reached on a 0.5 threshold; however, the maximum increase in the index is observed when switching from threshold 0.2 to 0.3, with very small increases in the index for larger thresholds. A probability threshold of 0.3 ensures a 98.61% true positive rate, with the cost of a 8.90% false discovery rate, and it is therefore the chosen value in the current analysis.

In light of the presented results, with the goal of identifying the anomalous cases of changes in assets' IDs between two quarters, assuming that the percentage of such changes in a quarter is 5% of all reported assets, a random forest model results as the best choice. In fact, among the tested models, the random forest ensures large accuracy and balanced accuracy. Moreover, selecting a probability threshold for classification of 0.3, the best model provides a true positive rate around 99% and a cost of a less than 9% in terms of false discovery rate that is considered acceptable in the DQM process.

4.2 Asset type-detailed results

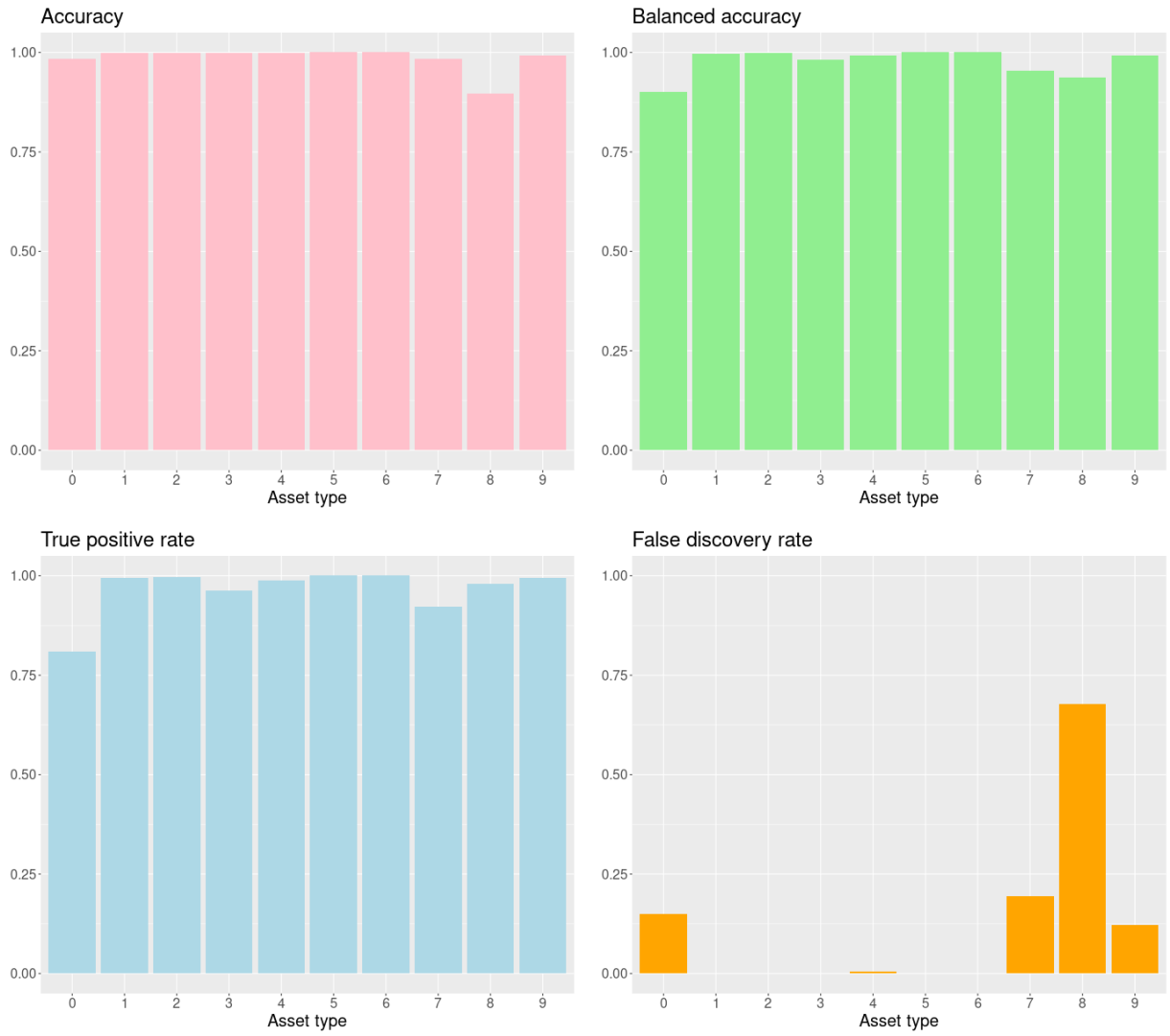
As previously mentioned in the introduction, in this paper we also conduct a deeper analysis to evaluate the performance of the selected best model[13] on different "asset types".

Figure 3 below shows the results, in terms of accuracy, balanced accuracy, true positive rate and false discovery rate, for an unbalance percentage of $p = 5\%$, with the previously chosen general probability threshold 0.3. As observable, performance do vary depending on the asset type: in fact, securities (asset type from 1 to 6) have very good results for all four metrics. Differently, false discovery rates for non-securities assets (asset type 0, 7, 8 and 9) are quite high, especially for asset type 8 (loans); this means that we can expect that many of the predicted cases of matches would actually be erroneously classified. Despite this cost in terms of FDR, true

[13] The random forest with ntrees=100, mtry=5 and a probability threshold of 0.3.

positive rates are quite high also for non-securities assets, proving that the model continues being effective in these cases.

Figure 3. Accuracy, balanced accuracy, TPR and FDR for different asset types.



As explained, security asset types (1-6) show excellent results, while lower performance is observed for non-security asset types (0, 7-9). A first possible reason that could explain this variability in performance is the different volume of data in each asset type class in the database, being small for non-securities; a second motivation could be the presence or absence of standard for the identification code: ISIN codes are widely available for securities and reporting by standard is strongly recommended by regulators. On the other side, IDs for assets that are not securities are often chosen arbitrarily by reporting insurance corporations. Finally, a third reason could be that available features for non-securities assets are less informative and many features used in the training matrix are securities-specific (e.g. issue date, issuer sector, issuer area).

Given the observed large values for the FDR in asset types 0, 7, 8 and 9, a more in depth analysis on these four cases can be held: in order to improve performance of the model on these four classes, a class-specific probability threshold can be investigated, since the globally optimal threshold of 0.3 is not performing sufficiently well.

In Figures 4 and Tables 4 are presented the model performance metrics TPR and FDR for the types of assets 0, 7, 8 and 9, varying with the probability threshold for classification. We will now focus on each type of asset separately, in order to choose the best probability threshold in each case.

Figure 4.1 Model performance measures (TPR, FDR, TPR-FDR) for asset type 0

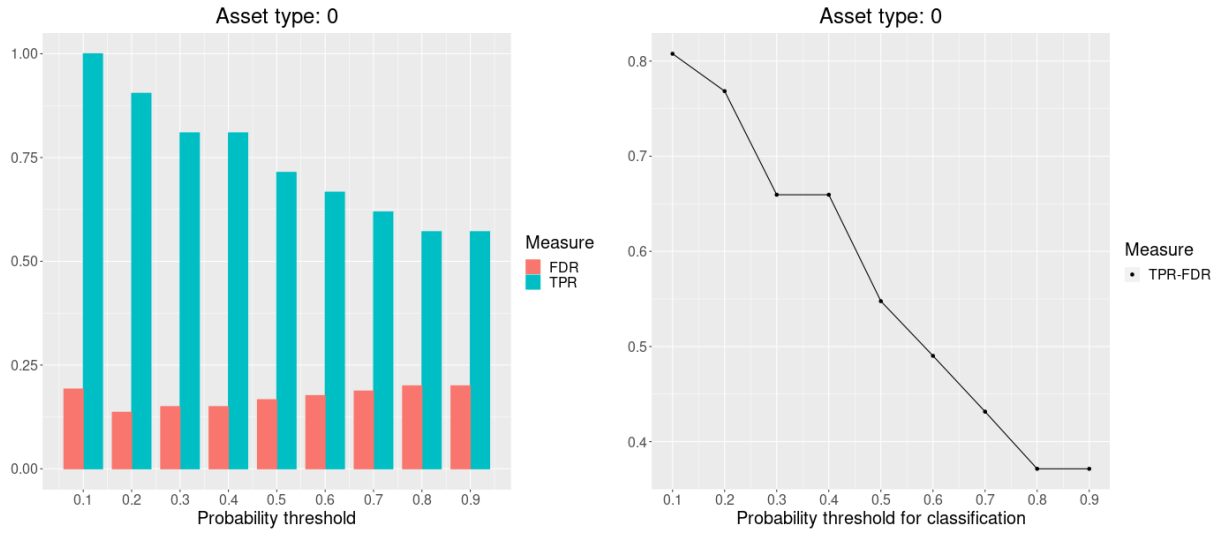


Table 4.1 Model performance measures for asset type 0

	Asset type: 0								
Threshold	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
Accuracy	98.81	98.81	98.33	98.33	97.86	97.62	97.38	97.14	97.14
Balanced accuracy	99.37	94.86	90.10	90.10	85.34	82.96	80.58	78.20	78.20
TPR	100.00	90.48	80.95	80.95	71.43	66.67	61.90	57.14	57.14
FDR	19.23	13.64	15.00	15.00	16.67	17.65	18.75	20.00	20.00
TPR-FDR	80.77	76.84	65.95	65.95	54.76	49.02	43.15	37.14	37.14

As observable from Figure 4.1 on the left, TPR and FDR have two different trends with the increasing probability threshold; the difference between TPR and FDR decreases. As observable from the plot on the right, performance for the asset type 0 can be improved by choosing a smaller probability threshold than the general value of 0.3. In fact, choosing a threshold of **0.2**, TPR reaches 90.5 and FDR equals 13.6. Although a smaller threshold of 0.1 would provide a TPR of 100, this would lead to an excessively high FDR (19.2).

Figure 4.2 Model performance measures (TPR, FDR, TPR-FDR) for asset type 7

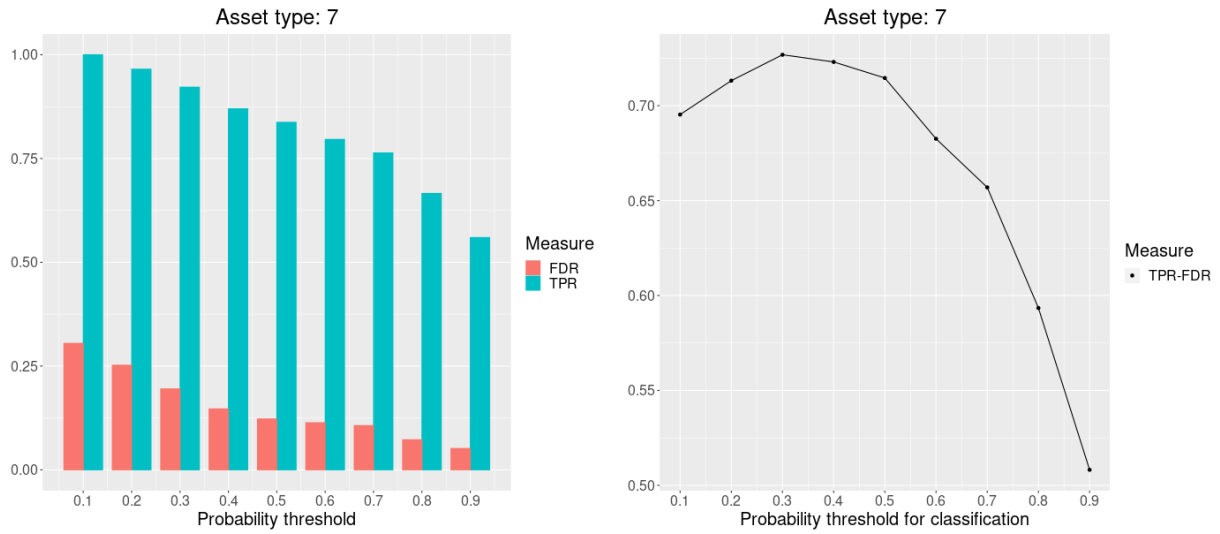


Table 4.2 Model performance measures for asset type 7

	Asset type: 7								
Threshold	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
Accuracy	97.81	98.20	98.49	98.60	98.60	98.47	98.36	98.07	97.64
Balanced accuracy	98.85	97.41	95.51	93.10	91.56	89.54	87.94	83.16	77.90
TPR	100.00	96.53	92.19	86.98	83.73	79.61	76.36	66.59	55.97
FDR	30.47	25.21	19.51	14.68	12.27	11.35	10.66	7.25	5.15
TPR-FDR	69.53	71.32	72.68	72.30	71.46	68.26	65.70	59.34	50.82

As observable from the Figure 4.2 on the left, TPR and FDR both decrease with increasing probability threshold, although with different speed. Indeed, the difference between the two metrics, observable in the plot on the right, reaches its maximum for a threshold of 0.3. Even though threshold 0.3 does provide a 92.2 TPR, the FDR is considered to be too large (19.5); therefore, selecting a larger threshold of **0.4**, a satisfying 87 TPR is reached with a 14.7 FDR, which is preferable.

Figure 4.3 Model performance measures (TPR, FDR, TPR-FDR) for asset type 8

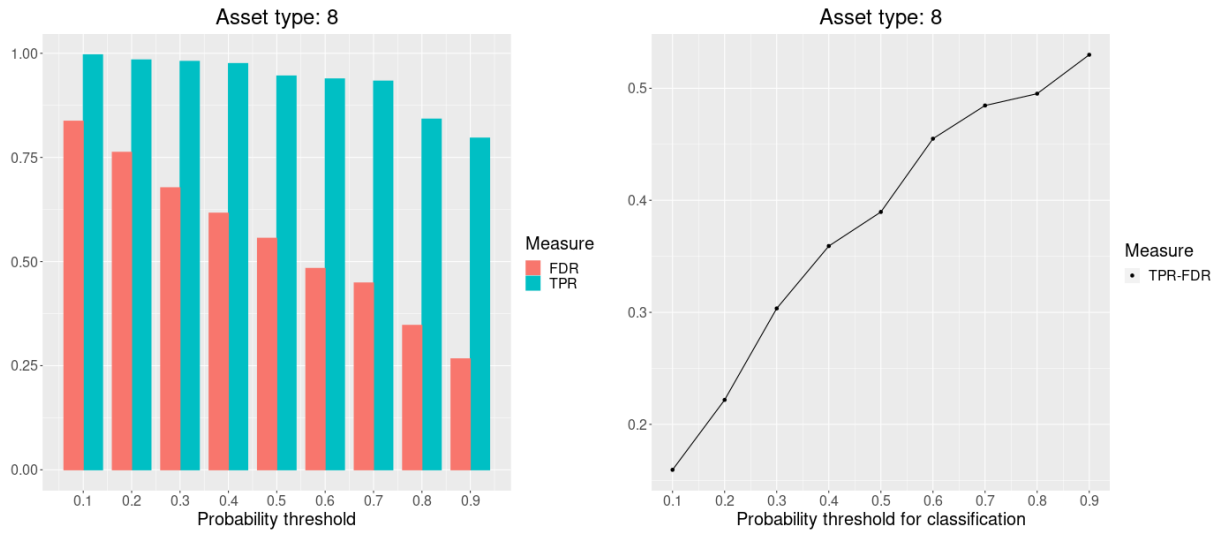


Table 4.3 Model performance measures for asset type 8

	Asset type: 8								
Threshold	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
Accuracy	74.34	84.11	89.60	92.03	93.80	95.29	95.86	96.97	97.53
Balanced accuracy	86.33	90.89	93.61	94.64	94.16	94.61	94.66	90.93	89.06
TPR	99.65	98.42	98.07	97.54	94.56	93.86	93.33	84.21	79.65
FDR	83.71	76.24	67.73	61.63	55.60	48.36	44.87	34.69	26.66
TPR-FDR	15.94	22.18	30.35	35.92	38.96	45.50	48.46	49.52	52.99

As observable from Figure 4.3 on the left, TPR and FDR both decrease with increasing probability threshold, although with different speed: FDR decreases faster than TPR. Consequently, the difference between the two metrics does increase. Even though threshold 0.3 does provide a 98.1 TPR, the FDR is considered extremely large (67.7) and not acceptable; therefore, selecting a larger threshold of **0.8**, a satisfying 84.2 TPR is reached with a 34.7 FDR, which is still large but provides a good compromise.

Figure 4.4 Model performance measures (TPR, FDR, TPR-FDR) for asset type 9

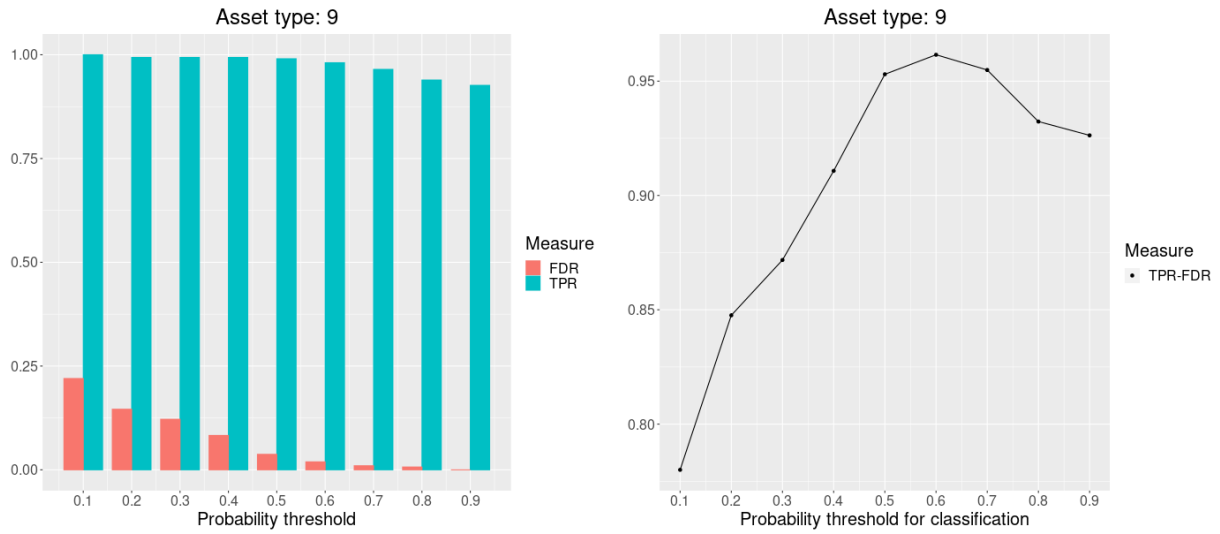


Table 4.4 Model performance measures for asset type 9

	Asset type: 9								
Threshold	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
Accuracy	98.59	99.12	99.28	99.52	99.76	99.81	99.78	99.66	99.63
Balanced accuracy	99.26	99.23	99.32	99.44	99.42	98.99	98.21	96.94	96.31
TPR	100.00	99.36	99.36	99.36	99.04	98.08	96.47	93.91	92.63
FDR	22.00	14.60	12.18	8.28	3.74	1.92	0.99	0.68	0.00
TPR-FDR	78.00	84.76	87.18	91.07	95.30	96.15	95.49	93.23	92.63

As observable from Figure 4.4 on the left, TPR and FDR both decrease with the increasing probability threshold; the difference between TPR and FDR reaches its maximum for a threshold of 0.6; as observable from Table 4.4, choosing this value does provide a large TPR (98.1) with a very low FDR (1.9). For this reason, threshold **0.6** can be chosen in the model for asset type 9.

In conclusion, the chosen values for the probability threshold for each asset type are reported in Table 5.

Table 5. Fine-tuned probability threshold for each asset type.

Asset type	**Selected probability threshold**
0	0.2
1-6	0.3
7	0.4
8	0.8
9	0.6

5. Conclusions

Errors in insurance reporting may lead to unexpected and undesirable changes in the IDs of the reported assets from one quarter to the next. An automated method to detect such changes is necessary to improve the data quality of insurance statistics, which are published at the international level, given the volume of the available data, the level of granularity for the single assets and the non-negligible impact of these changes on compiled statistics.

A record linkage approach using supervised machine learning classification models is proposed to achieve this goal.

Real quarterly Italian data from 2019-2022 are used for the application. Three models are considered, i.e. the logit model as a benchmark, random forests and neural networks as machine learning options. Robust results are presented, testing the models on differently sampled data, stratified by different percentages of code change cases in two consecutive quarters, since the true proportion of such anomalies in the data is currently not known with precision. Moreover, the robustness of the test results is shown for all the different models and periods considered.

The tested models performed well in terms of average metrics (AUC) and the results show the superiority of the random forest model to approach the problem compared with the other tested classifiers (logit and neural networks).

Taking a 5% proportion of anomalies in the data and a 0.3 probability threshold for classification, the selected random forest model shows good overall performance in all measures, both in terms of effectiveness and efficiency, ensuring both high accuracy and balanced accuracy, with a rate of correctly identified cases of ID changes (true positive rate) of around 99%, accepting the cost of a false discovery rate that approaches 9%.

Looking more closely at the performance of the selected random forest model, the test results for the different asset types are heterogeneous: the performance is lower for non-security asset types, mainly due to the small volumes of data, the lack of a standard for building IDs and the limited information held in the features used for classification. For this reason, different probability thresholds are selected for each non-security asset type to improve performance.

The presented test results provide an estimate of the improvement in data quality that would result from running the selected model on production data, with the goal of successfully identifying cases of unexpected and erroneous changes in IDs.

However, the estimated cases of change need to be validated by cross-checking with the insurance corporations in order to measure the actual//real performance of the proposed methodology. To this end, the model was used during four real production rounds in 2022 and proved to be effective: the first feedback from the check

with the insurance corporations, obtained thanks to the collaboration of IVASS, showed that around 60% of the detected cases are indeed anomalouswhile feedback on the remaining 40% is still under investigation.

Finally, the model should be periodically monitored in the future and its parameters updated if performance deteriorates during production rounds. However, these updates are not expected to be frequent, as the results presented in this paper are robust for all the couples of quarters considered.

References

Benjamini, Y., Drai, D., Elmer, G., Kafkafi, N., Golani, I. (2001) "Controlling the false discovery rate in behavior genetics research", Behavioural Brain Research 125, 279–284.

Bishop, C. M. (1995), "Neural Networks for Pattern Recognition", Oxford University Press.

Boriah, S., Chandola, V., Kumar, V. (2008), "Similarity Measures for Categorical Data: A Comparative Evaluation", SIAM International Conference on Data Mining.

Breiman, L. (2001), "Random Forests", Machine Learning, 45, 5-32.

Breiman, L. Bagging (1996), "Predictors", Machine Learning, 24, 123-140.

Buzzi, M. R., Costanzo, G., Di Lucido, M., La Ganga, B., Maddaloni, P., Svezia, E., Zambuto, F., Papale, F. (2020), "Quality checks on granular banking data: an experimental approach based on machine learning", Questioni di Economia e Finanza 547.

Chakraborty C., Joseph A. (2017), "Machine Learning at Central Banks", Bank of England Staff Working Paper, No. 674.

Cusano, F., Marinelli, G., Piermattei, S. (2021), "Learning from revisions: a tool for detecting potential errors in banks' balance sheet statistical reporting", Questioni di Economia e Finanza 611.

D'Orazio, M., Di Zio, M., Scanu, M. (2006), "Statistical Matching, Theory and Practice", Wiley.

Feigenbaum, J. (2016), "A Machine Learning Approach to Census Record Linkage", Working paper.

Fellegi, I., Sunter, A. (1969), "A theory for record linkage", Dominion Bureau of Statistics.

Ferrie, J. P. (1996), "A new sample of males linked from the public use micro sample of the 1850 U.S. Federal Census of Population to the 1860 U.S. Federal Census manuscript schedule", Historical methods: a journal of quantitative and interdisciplinary history, Vol. 29, 141-156.

Fortini, M., Liseo, B., Nuccitelli, A., and Scanu, M. (2001), "On Bayesian record linkage", Research in Official Statistics, 4: 185–198.

Jaro, M. (1989), "Advances in record-linkage methodology as applied to matching the 1985 census of Tampa, Florida", Journal of the American Statistical Association, 84: 414-420.

La Serra, V., Svezia, E. (2022) “Statistical matching for anomaly detection in insurance assets granular reporting”, Irving Fisher Committee (IFC) Working Papers of the Bank for International Settlements, no. 22, 11 October 2022.

Larsen, M. D., Rubin, D. B. (2001), “Iterative automated record linkage using mixture models”, Journal of the American statistical association. 96:453, 31-41.

Maddaloni, P., Continanza, D. N., del Monaco, A., Figoli, D., di Lucido, M., Quarta, F., Turturiello, G. (2022), “Stacking machine-learning models for anomaly detection: comparing AnaCredit to other banking datasets”, Questioni di Economia e Finanza, n. 689.

Okner B. (1972), “Constructing a New Data Base from Existing Microdata”, Annals of Economic and Social Measurement, Volume 1, number 3.

Rijpma, A., Cilliers, J., Fourie, J. (2020), “Record linkage in the Cape of Good Hope Panel”, Historical methods, vol. 53, no. 2, 112-129.

Rosenwaike I., Hill, M. E., Preston, S. H., Elo, I. T. (1998), “Linking death certificates to early census records: the African American matched records sample”, Historical methods: a journal of quantitative and interdisciplinary history. Vol. 31, 65-74.

Ruggles, S. (2002), “Linking historical censuses: a new approach”, History and computing, vol. 14, 213-224 (2002).

Tancredi, A. and Liseo, B. (2011), “A hierarchical Bayesian approach to record linkage and population size problems”, Annals of Applied Statistics, 5: 1553–1585.

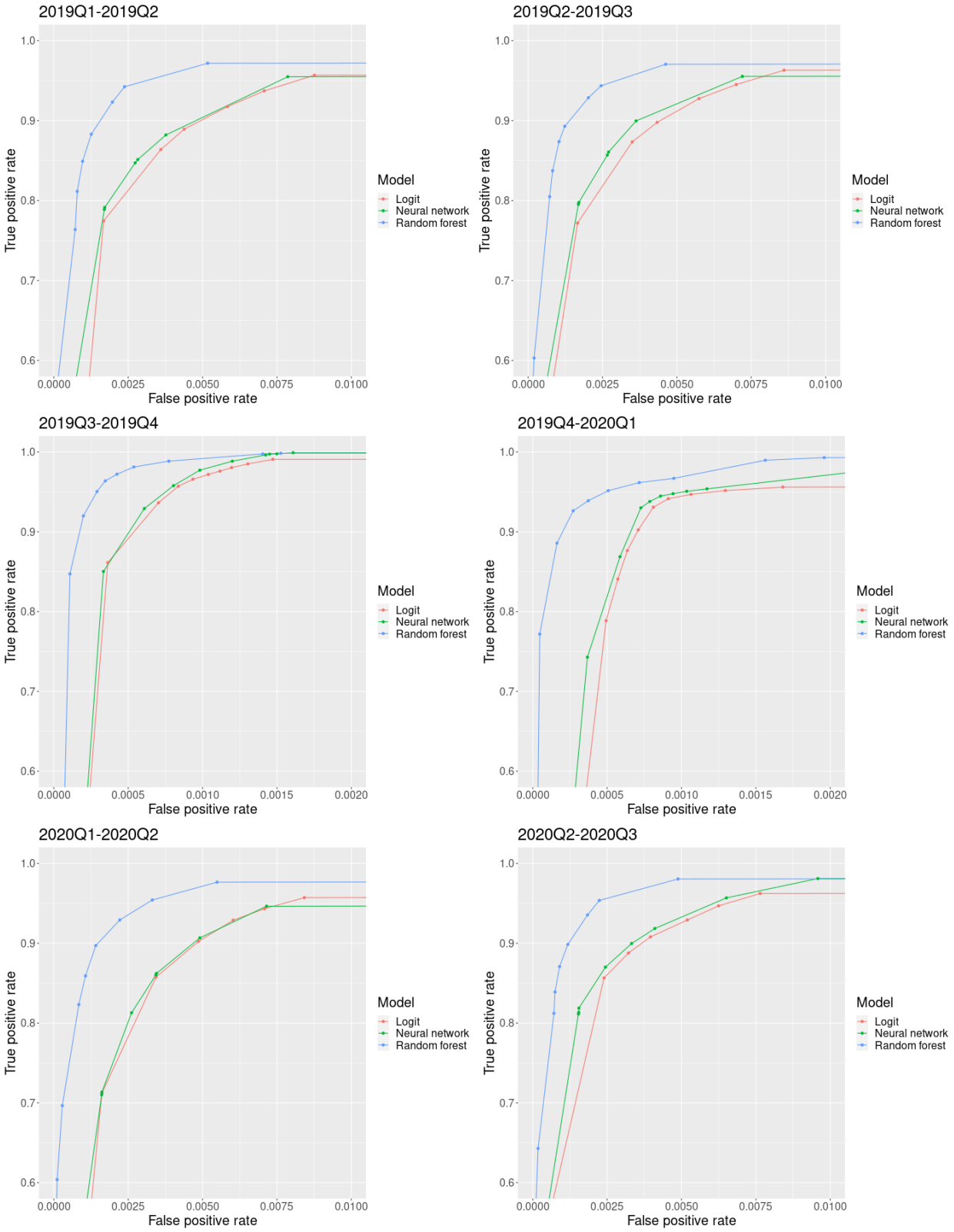
Vatsalan, D., Christen, P., Verykios, V. S. (2013) “A taxonomy of privacy-preserving record linkage techniques”, Journal of Information Systems (JIS), vol. 38, no. 6, pp. 946–969.

Zambuto, F., Arcuti, S., Sabatini, R., Zambuto, D. (2021) “Application of classification algorithms for the assessment of confirmation to quality remarks”, Questioni di Economia e Finanza 631.

Egmont-Petersen, M., de Ridder, D., Handels, H. (2002) “Image processing with neural networks – a review,”, Pattern Recognition, Vol. 35, No. 10, pp. 2279-2301,

Ossama, A.-H, Abdel-rahman, M., Jiang, H., Deng, L., Penn, G., Yu, D. (2014) “Convolutional Neural Networks for Speech Recognition”, IEEE/ACM Transactions on Audio, Speech and Language Processing, vol. 22, no. 10.

Appendix


2019Q1-2019Q2
2019Q2-2019Q3
2019Q3-2019Q4
2019Q4-2020Q1
2020Q1-2020Q2
2020Q2-2020Q3
True positive rate
False positive rate
Model
Logit
Neural network
Random forest

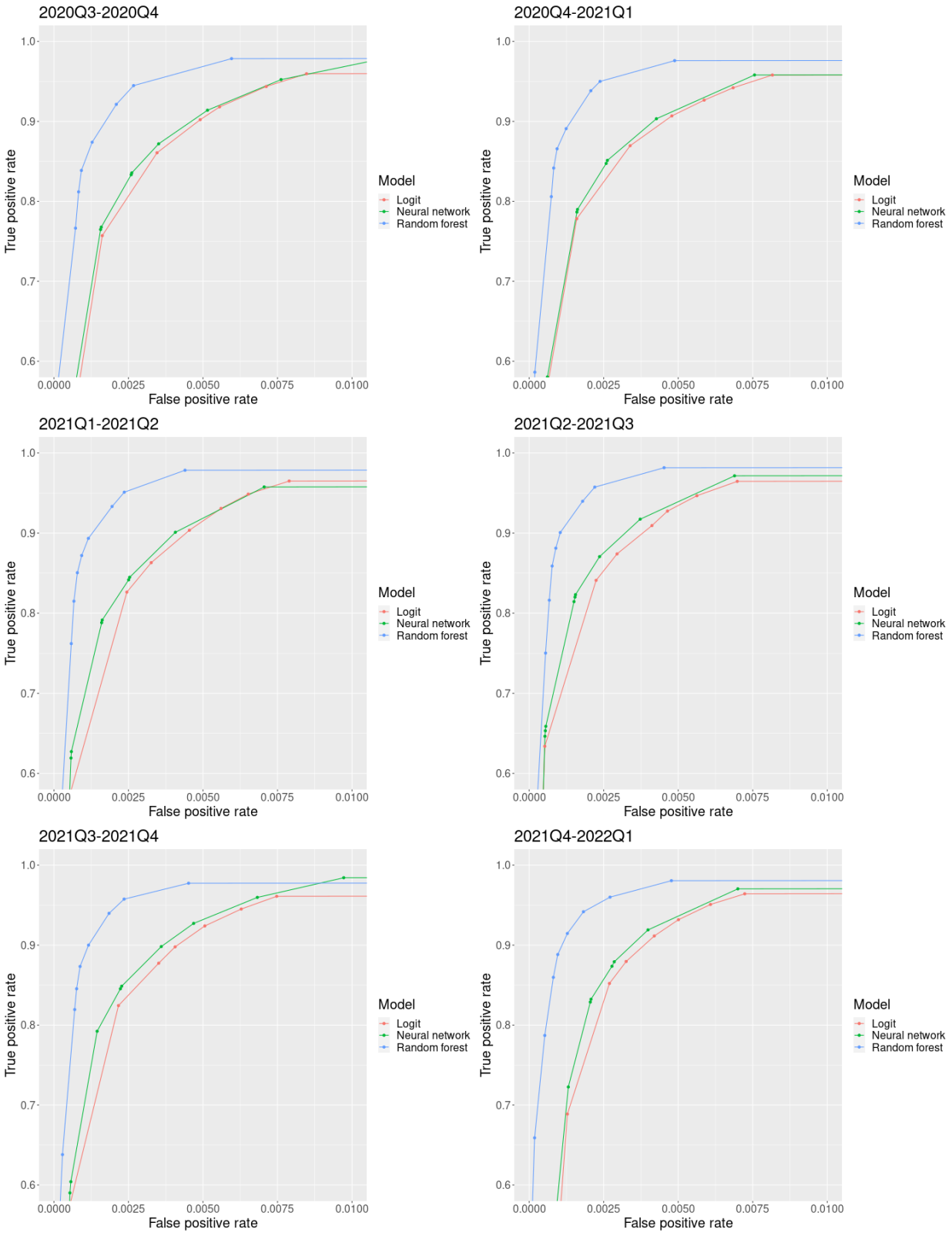

2020Q3-2020Q4
2020Q4-2021Q1
2021Q1-2021Q2
2021Q2-2021Q3
2021Q3-2021Q4
2021Q4-2022Q1
True positive rate
False positive rate
1.0
0.9
0.8
0.7
0.6
0.0000
0.0025
0.0050
0.0075
0.0100
Model
Logit
Neural network
Random forest

Table A.1 A confusion matrix

	Predicted positive	Predicted negative
True status: positive	A	B
True status: negative	C	D

Table A.2 Performance metrics of a binary classification supervised model.

Metric	Formula
TPR	A/(A+B)
TNR	D/(C+D)
FPR	C/(C+D)
FDR	C/(A+C)
Accuracy	(A+D)/(A+B+C+D)
Balanced accuracy	(TPR+TNR)/2

Table A.3 AUC index of the ROC curves for the different percentages of unbalance for the quarters 2021Q4-2022Q1.

AUC (%)	Unbalance
98.99	1%
99.57	2%
99.82	3%
99.92	4%
99.89	5%
99.90	6%
99.92	7%
99.91	8%
99.90	9%
99.89	10%
99.88	20%
99.87	30%
99.85	40%
99.79	50%
99.79	Average

Glossary

IC: insurance corporation

ID: identification code of the asset

DQM: data quality management

TPR: true positive rate

TNR: true negative rate

FDR: false discovery rate

FPR: false positive rate

AUC: area under curve

ROC: receiver operating characteristic