AI-Driven Data Quality and DataOps Management

Anonymous authors
Anonymous Affiliation

Abstract

Data Operations (DataOps) [16], a crucial field in AI, has also been adopted in finance to automate routine processes and improve operations. We highlight the importance of data quality management in financial services, where data is often incomplete, inconsistent, and contains duplicates, outliers and missing values. Additionally, applications of two types of data Quality Control (QC) are shown; Rule-Based QC which involves setting predefined rules for data validation and Machine Learning (ML) based QC which uses predictive algorithms and anomaly detection to identify inaccurate and incorrect data points. DataOps further enhance QC by automating and streamlining data pipelines enabling continuous monitoring and quick identification of errors thus improving overall operational efficiency. We propose a platform-agnostic, scalable and customizable Python-based Advanced QC framework for performing data quality checks and anomaly detection. We illustrate how the Advanced Data QC framework can be used on publicly available financial datasets and showcase anomaly detection algorithms using ADBench data. The framework is designed to ensure data accuracy, consistency, and completeness, which is essential for meaningful analytics, predictive modelling and decision-making. This paper is an excellent reference for anyone looking to implement this framework on any internal organizational data and enhance data quality within their organization or processes from the ground up.

Keywords

Data, DataOps, QC, ML-Based, Rule-Based, AI.

ACM Reference Format:
Anonymous authors. 2024. AI-Driven Data Quality and DataOps Management. In *Proceedings of 5TH ACM International Conference On AI in Fianance (ACAIF '24).* ACM, New York, NY, USA, 9 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

In the rapidly evolving financial industry, the integration of DataOps and the assurance of data quality have become paramount. Financial institutions are increasingly reliant on vast amounts of data to drive decision-making, enhance customer experiences, and maintain regulatory compliance.

ACAIF '24, October 20–31, 2024, Woodstock, NY

ACM ISBN 978-1-4503-XXXX-X/18/06
https://doi.org/XXXXXXX.XXXXXXX

1.1 About Data

"Data" can be defined as any set of values, facts, or figures that can be processed, analyzed, and used for decision-making. It includes numerical, textual, and categorical information and can be generated through various means such as observations, measurements, or transactions. In the financial services sector, data refers to any information used to support financial decision-making, including historical data, current data, and projections. This data can be reported periodically, such as on a monthly, quarterly, or annual basis, and is utilized for risk assessments, performance evaluations, or compliance reporting, etc. [29]. Financial Institutions are burdened with enormous amounts of data and the data manifests in various forms, including financial reports (balance sheets, profit and loss statements, cash flow reports), market data (real-time or historical prices, indices, market transactions), client and transaction data (personal data, transaction histories, credit ratings), technical documents (blueprints, software specifications) and risk and compliance data (metadata, lineage, regulatory reporting) [33]. These diverse data types are crucial for internal decision-making, regulatory compliance, trading, risk assessment, portfolio management, financial planning, and governance.

For the scope of this paper, we will focus on tabular and streaming tabular data. Tabular data is structured and organized in rows and columns, typically found in databases and spreadsheets, data warehouses [25, 34]. Streaming tabular data, on the other hand, is data that is continuously generated and processed in real-time, such as stock market feeds, transaction logs, news articles, etc. [3, 34]. Data in the financial industry can originate from diverse sources including outputs of predictive models, stock exchanges, central banks, third-party vendors such as news agencies, market data and proprietary databases. Data QC can be applied either on the model inputs or outputs.

- Model Outputs: Data generated by financial models must undergo QC to ensure accuracy and reliability before being used for decision-making or reporting [31].
- Third-Party Vendors: Data acquired from external sources should be validated and verified to maintain consistency and trustworthiness [17].

To build a state-of-the-art data quality framework, it's essential to explore key questions such as, can a model be classified as data? A model is a quantitative method that processes input data into quantitative estimates using statistical, economic, financial, or mathematical theories [29]. Models are crucial for risk management but require careful validation to manage model risk, which can lead to financial loss if models underperform [35]. Although models are constructed from data, their parameters and structure can be considered metadata, requiring quality assurance. Then, does a model's output qualify as data? Yes, model outputs are valuable new data points for further analysis, predictions, or decision-making. Lastly, defining the true origin of data for model outputs includes both the

input data and the model itself. Financial institutions often use processed data that is incomplete, inconsistent or contains duplicates, leading to flawed analytics and decision-making. Therefore, data quality checks are critical for building reliable machine-learning models [1, 2].

1.2 Applications of Data Quality Control

When a model's output serves as input data for another model, it is essential to distinguish between model surveillance and data quality control (QC) from an organizational perspective:

- Model Diagnostics/Model Enhancement Framework: Identifying specific data subsets where the model underperforms and analyzing feature distribution differences to pinpoint causes. Addressing data issues and model bias by targeting problematic features and experimenting with alternative models. Evaluating and comparing multiple models to select the best one, justifying advanced techniques like model ensemble based on varying prediction accuracies.
- Model Surveillance: Continuous monitoring of model performance is necessary to detect any deviations or anomalies. This includes tracking key performance indicators (KPIs) and ensuring that the model remains robust over time.
- Analytical Use Case: Data QC is critical when the model's output is used for analytical purposes. Ensuring the accuracy and reliability of this data helps in making informed decisions and deriving meaningful insights.
- Charts and Visualizations: Visual representations of data and model outputs can help identify trends, patterns and anomalies. High-quality data ensures that these visualizations are accurate and actionable.
- Other applications include risk modelling, stress testing, and scenario analysis, especially under Basel III regulations, fraud detection.

1.3 Stages of Data Quality Control, Methodology and Best Practices

Data QC is crucial at multiple stages: Input Data Stage (validating raw and third-party data), Model Output Stage (ensuring outputs are error-free for subsequent models), and Post-Processing Stage (verifying results for accuracy and reliability) as shown in Figure 1 below. The initial step in constructing a model involves acquiring data, establishing basic quality control measures, and collaborating with the upstream team responsible for data provision. Automating these quality control steps is crucial, as quality control and model surveillance are interrelated and inseparable.

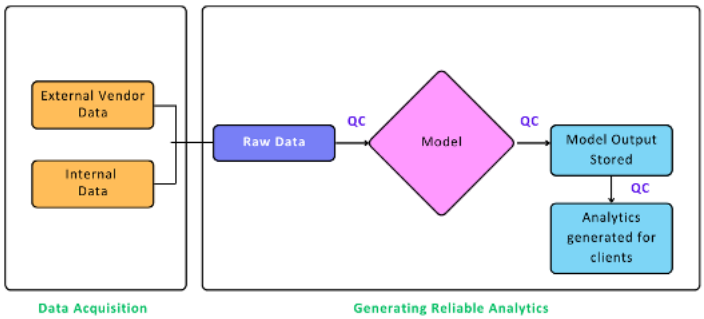


Figure 1: Data QC Flow Chart

A robust quality control process allows differentiation between model performance issues caused by data problems and those due to market condition changes, enabling appropriate actions. Providing a confidence score for the model further enhances its reliability, helping users gauge the trustworthiness of the model's predictions. Incorporating interpretability techniques into the model fosters transparency, understanding, and trust, particularly with complex machine learning models, facilitating their effective deployment in real-world scenarios. Establishing end-to-end standards and procedures from data acquisition through ETL processes and data provisioning is essential to streamline the quality control process within the organization. Clear handovers and communication channels between teams ensure data is delivered consistently and on time, maintaining high-quality standards throughout.
In the following sections, we explore various data quality control techniques, including rule-based and ML-based QC approaches. We focus on the Advanced QC library, a generic framework built in compliance with the National Institute of Standards and Technology [27] for data quality checks and data profiling. The framework is designed to enhance data governance for standard or advanced AI models. It has been built to leverage TensorFlow Data Validation [36] library for rule-based QC and Python Outlier Detection library (PyOD) [39] for ML-based QC, adding custom generic functions developed internally to cater for various QC requirements.

2 Literature Review

Data quality is defined by several dimensions, including accuracy, completeness, consistency, and timeliness [22]. Ensuring these dimensions of data quality are met is paramount for effective decision-making in organizations, particularly in sectors where regulatory oversight is stringent like in the financial services industry, regulatory frameworks mandate high standards for data accuracy and reporting. According to Gartner's 2017 Data Quality Market Survey, organizations have attributed an estimated $15 trillion in annual losses to flawed data [23]. Additionally, a 2024 AI survey [37] highlighted that poor data quality or inaccurate data leads to an average loss of 6% in revenues for companies. This increases the need to have data governance structures in place to guarantee data used for training models and performing various analytics produces reliable insights. Increased reliance on data-driven technologies creates a more urgent need for reliable data specifically when insights derived from the data inform operational strategy,

improve decision-making using predictive analytics and help with regulatory compliance.

The need to ensure the reliability and dependability of AI systems before they are widely adopted has led to the emergence of AI factories. AI factory is an AI-centered decision-making engine that optimizes daily operations by enhancing small-scale decisions using machine learning algorithms [20]. The core elements in AI factories include; data pipelines, algorithm development, experimentation platforms and software infrastructure. Data pipelines can be integrated into data QC as the component involves collecting, processing and analyzing data by gathering, cleaning, integrating processing and safeguarding data [20]. Several companies have adopted AI factories in their process to refine customer experiences through optimized real-time data analyses leading to improved data-driven insights. Such companies include Netflix, Uber, and Google [15]. Data-centric AI, which is a growing trend in AI has placed data at the forefront of AI development. Researchers and practitioners have gradually focused their efforts on enhancing data quality [8]. This shift underscores the critical role data QC plays in the AI pipelines. Extensive research has demonstrated the profound impact of data quality on the performance and reliance of ML algorithms. In recent years, Google developed an open-source data validation tool - TensorFlow Data Validation (TFDV) used in ML pipelines [6]. This tool has greatly been adopted by other organizations and forms an integral basis for the advanced QC framework discussed in this paper. Research by [32] highlights the importance of data validation for ML applications considering how incorrect or missing data can greatly impact decisions made in downstream processes.

Data cleansing is a fundamental component of evaluating data governance, which focuses on quality and is an essential step before creating data analytics [38]. Literature by Chu et al. [7] demonstrates automated methods for detecting and fixing data errors. Such methods can be utilized in rule-based QC checks. It is imperative to understand the criteria being used to evaluate the health of a dataset, therefore a set of constraint-based rules is critical to evaluate whether the bounds within which rules are set are met or not. By combining rule-based checks with ML algorithms, financial institutions can create hybrid models that maximize the strengths of both approaches, thus improving data reliability and operational efficiency.

Rule-based QC involves predefined rules and criteria to ensure data integrity through cleaning, validation, standardizing data and identifying discrepancies in datasets. Polyzotis et al. [30] demonstrates Implementations using TFDV for rule-based QC checks including detecting errors in data, identifying skewness and data drift [36]. ML-based QC leverages algorithms to learn from historical data and identify anomalies or patterns indicative of data quality issues. ML-based QC approach has garnered attention for its ability to handle complex datasets and adapt to evolving data quality challenges. Recent studies indicate that ML approaches can enhance QC by automating the detection of errors and discrepancies in data before these errors are propagated into further downstream processes [10, 30].

The automation presented by ML algorithms has eased the manual process of detecting outliers and immensely reduced the risk posed by using erroneous data [26]. For ML models to be reliable, the user needs to ensure the input data is free of errors, check if there is a need to retrain the model once new data is introduced and whether data used for predictive analysis is significantly different from the data used to train the model initially [10]. Anomaly detection using machine learning algorithms involves identifying unusual patterns within a dataset that deviate significantly from the norm [5] based on; labelled data for supervised models, unlabeled data using unsupervised models and semi-supervised models that assume labelled data on normal data points only thus marking anything unordinary as anomalous [24]. These instances that are different from the rest of the data points help indicate errors or fraud.

Considering the growing need for organizations to develop comprehensive data quality management strategies that ensure regulatory compliance and data integrity, we demonstrate the applications of different ML models for anomaly detection using sample publicly available datasets, as highlighted in Section 5 below. Additionally, the paper illustrates different rule-based QC approaches that have been developed into a library for identifying common data problems detailed in section 3. Components within this library have also been configured to detect different data anomalies such as skewness, detecting data drift by checking series of data and performing validity checks by comparing rules specified by a user and the statistics of the input data, expanding the implementation by Google [6], using TFDV for data validation.

3 Types of Data QC

We explore the two main types of Data Quality Control (QC): Rule-based and ML-based. Rule-based QC employs predefined criteria to validate data integrity systematically and is the first component of the Advanced QC Framework library, which handles various data formats and uses the TFDV library to identify outliers by comparing statistics against a schema [36]. Customized reports are generated for any detected breaches, and the framework also evaluates data for drift and skewness to ensure model reliability. ML-based QC leverages algorithms for anomaly detection and missing data imputation, improving data quality by identifying unusual patterns and maintaining data integrity [5]. ML algorithms fall into four categories: supervised, unsupervised, semi-supervised, and reinforcement learning [4]. PyOD, an open-source Python library, is highlighted for its scalability and ease of integration in ML workflows, providing over 50 anomaly detection algorithms [41]. Effective management of missing values is crucial, with approaches like imputation and using ML algorithms like Random Forest to handle missing data [11, 13]. Tables 1 and 2 (see Appendix.) describe the exhaustive list of rules and algorithms integrated into the framework.

4 Applications of Data Quality Control Methods

Data quality control plays a vital role in ensuring accuracy, reliability and consistency of data being used for upstream and downstream tasks, especially in financial services where the regulations are stringent. It is imperative to understand the criteria being used to evaluate the health of a dataset using rule-based validation, automated anomaly detection, detecting data discrepancies etc. By applying robust data quality controls, organizations can enhance operational efficiency, ensure compliance with regulations and drive

more accurate insights from their data. In the following segments, we highlight how the Advanced QC framework can be applied to any dataset to perform data profiling.

4.1 Advanced QC Using Rule-Based Data Quality Assessment

We illustrate the standard workflow of the advanced QC framework whereby the main functions can be amalgamated in a folder that can be developed into a back-end library. In the front-end code, a user can call the relevant functions to perform the QC checks depending on their requirements. Additionally, the paper demonstrates how ML algorithms can be applied for anomaly detection and how this can be integrated into an organization's data pipelines.

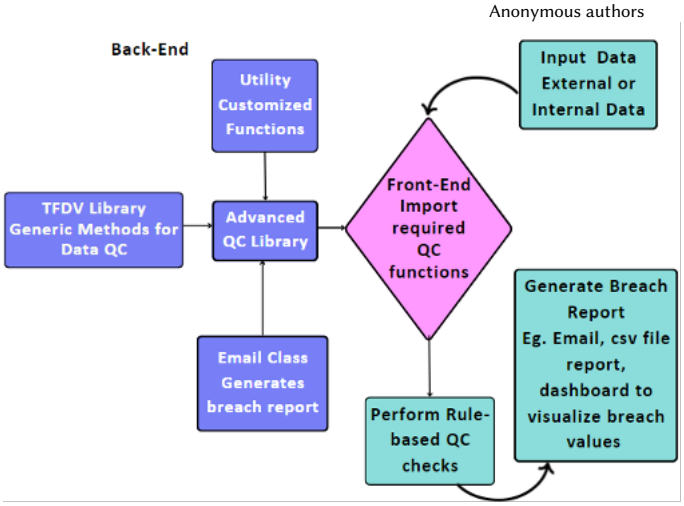


Figure 2: Advanced QC Rule-Based Data Quality Assessment Application Flow Chart

4.1.1 *Back-End Process.*

In the backend, the main functions can be amalgamated in a folder that can be developed into a Python library. This library can contain the main functions such as checking for duplicates, null values, missing column values, checking for skewness and data drift etc. as highlighted in Table 1 (see Appendix). At the core of the Advanced QC backend library, we can utilize the TensorFlow Data Validation library which is an open-source library that enhances identification of outliers in input data by comparing statistics against a schema python library [36]. Alternatively, one can utilize a customized configuration file that can include custom functions for data profiling, to be applied in the front end. Figure 2 below shows the workflow of how an advanced QC framework can be applied to perform standard and advanced QC checks.

4.1.2 *Front-End Process.*

In the front end, the relevant advanced QC functions can be imported from the custom Python library or can be called from the configuration file. Basic and advanced QC checks can be performed on raw data that can either be external such as vendor data or internal used for an organization's internal analytics processes. The bounds over which breach values should be flagged can be predefined in a config file in the front end. Once all checks are performed, depending on the output a user is interested in, one can leverage the email generation functionality present in the back-end library to generate a breach report. Additionally, the breach values can be stored in a file for further visualization using dashboards. Subsequently, this framework can be customized to evaluate relevant features from the input data frame to obtain basic statistical checks like count, mean, median, standard deviation, and percentage of zeros, for numeric features and count, unique, frequency, and string length, for categorical features. These statistics help give further insights and understand the distribution of each feature.

4.2 ML Models for Advanced QC

Machine learning models can be used to detect errors present in an organization's upstream or downstream data more effectively than traditional methods. Through learning from historical data, the ML algorithms can correct common errors, resolve outliers or highlight exceptions more accurately leading to more reliable input data used for downstream or upstream analytics processes and consequently more valuable insights. Considering the vast amounts of data an ML can handle at scale, incorporating ML algorithms for data QC is beneficial to an organization that has growing data needs thus reducing the manual execution of these and reducing human-prone errors. We demonstrate applications of PyOD which gives a range of algorithms that can be used to detect anomalies for time series data, graphical data, tabular data and distributed systemic data [41]. One must carefully select a correct PyOD model that suits your needs and depends on the type of data to be used for analysis.
Initially, data cleaning is performed to ensure the data is free from noise and biases before being fed into the anomaly detection model. Any features not suitable for modelling are then transformed by performing normalization, encoding categorical variables or scaling. The resulting data is then split into train and test randomly whereby the train set can be split further into train and validation datasets. Hyperparameter tuning is then carried out on the validation dataset to enhance the performance of the ML model. During model training for anomaly detection, an anomaly score is assigned for each data point depending on how far the data points deviate from the norm. Consequently, a threshold needs to be calibrated to distinguish between normal and anomalous data points. Finally, the model is evaluated using respective performance metrics on the test set, depending on the type of ML model used. Once anomalous data points are flagged, the user can resolve those breaches depending on the frequency of the data flow, either daily, weekly or monthly before the model is deployed, as shown in figure 3 below.

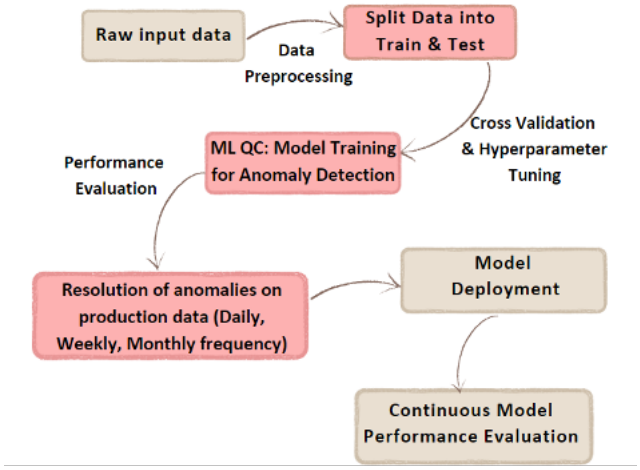


Figure 3: ML Based Data QC Flow Chart

5 Case Study

In real-world applications, Data QC should be created by model owners often in partnership with engineers to verify the accuracy of production analytics and all associated data sources regularly with predetermined (daily, weekly, monthly, etc.) frequency. This process may be automated and can be managed by a dedicated team to ensure consistency and scalability. In an automated QC framework, following the execution of each data or model job, a corresponding quality control job is triggered. If any threshold is breached during the quality control process, a notification is sent, and breach values are saved in a specific folder for review and resolution of data breaches. The subsequent job can only start once the previous quality control job has either run successfully or been manually signed off as a false alarm. The next job will run automatically, in the absence of breaches. We present a case study on the Rule-Based and ML-based QC checks performed on the "Corporate Bonds Indices" dataset on Kaggle [9] and one of the public ADBench benchmark datasets, Fraud Dataset.

5.1 Rule Based Data QC

We demonstrate the Advanced QC framework using rule-based QC by running several standard checks on the "Corporate Bonds Indices" dataset on Kaggle [9] consisting of various bond indices including corporate bonds from different sectors. It contains financial time series data (from 1998) which can be used to track the performance of bonds across multiple regions and sectors, making it a valuable resource for financial analysis, risk assessment, and performance monitoring. The checks included; "do_null_level" to calculate missing values in the data, "do_outlier_detection_std" to calculate lower and upper thresholds for outlier detection based on mean and standard deviation, flagging breaches that exceed these thresholds, "do_outlier_detection_range" to check for outliers if values lie outside the minimum and maximum values calculated based on ten historical data points for each column, "do_positive_only" to ensure no values are negative and "do_last_value_delta_check" to compare the latest data with the last available data and flag breaches if values remain constant throughout. The framework flagged a breach for "do_outlier_detection_range" only. The status of the checks and the functional status of each check is captured in a "Status File" as shown in Figure 4 below. It contains the run date of a particular check, check name, check run timestamp and the success status of the QC process.

Series Run Date	Check	Status Update Timestamp	Status
20111220	Missing Value Check	27/09/2024 15:03	Success - No Breach Detected
20111220	Positive Values Only	27/09/2024 15:03	Success - No Breach Detected
20111220	Outlier Check - Std-Dev Range	27/09/2024 15:03	Success - No Breach Detected
20111220	Outlier Check - Min-Max Range	27/09/2024 15:03	Success - Breach Detected
20111220	Value Delta Change Check	27/09/2024 15:04	Success - No Breach Detected
20111227	Missing Value Check	27/09/2024 14:01	Success - No Breach Detected
20111227	Positive Values Only	27/09/2024 14:01	Success - No Breach Detected
20111227	Outlier Check - Std-Dev Range	27/09/2024 14:01	Success - No Breach Detected
20111227	Outlier Check - Min-Max Range	27/09/2024 14:01	Success - Breach Detected
20111227	Value Delta Change Check	27/09/2024 14:01	Success - No Breach Detected
20111228	Missing Value Check	27/09/2024 13:57	Success - No Breach Detected
20111228	Positive Values Only	27/09/2024 13:57	Success - No Breach Detected
20111228	Outlier Check - Std-Dev Range	27/09/2024 13:57	Success - No Breach Detected
20111228	Outlier Check - Min-Max Range	27/09/2024 13:57	Success - Breach Detected
20111228	Value Delta Change Check	27/09/2024 13:57	Success - No Breach Detected

Figure 4: Status File representing the check run date, name, status-update timestamp and final status

Additionally, a report is generated to inform the relevant audience about the breach, including details such as the breach capture, its location in the flat file system, the input data location and breach values for a particular column. The "BREAK_THE_PROCESS" flag was set to false, allowing the process to continue with a yellow status. If "BREAK_THE_PROCESS" flag were set to true, the process would have broken until a data team investigates the breach and resolves it, then allowing the process to continue with a red status.

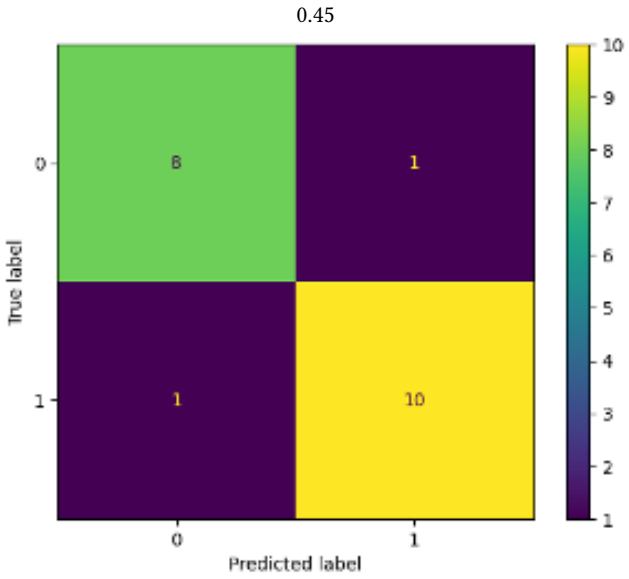
5.2 Advanced ML-Based Data QC

Fundamentally, the PyOD library has been integrated into the Advanced QC framework to perform outlier detection on downstream data with benchmarking done using ADBench datasets to verify the performance of the anomaly algorithms used. The ADBench Benchmark Dataset [12] is a widely used benchmark dataset for anomaly detection tasks. It is designed to test and evaluate various QC and anomaly detection techniques in different domains, including financial data quality assessments. We demonstrate the usage of supervised and unsupervised ML methods to detect outliers and use the Fraud dataset [18] to verify the results. The Fraud dataset contains about 284807 samples with 29 features and only 0.17% (492) anomalies; making it a heavily imbalanced dataset.

For illustration purposes, we randomly sampled an equal number of instances (100) from each class in the dataset, shuffled the data and extracted the resulting features and labels. The Extreme Boosting Outlier Detection (XGBOD) [40] model was then used to perform anomaly detection. We selected XGBOD because it combines the strengths of both supervised and unsupervised machine learning methods creating a hybrid approach which uses each of their capabilities to perform outlier detection. A simple grid search is performed using GridSearchCV [28] to tune hyperparameters. The training data was split into 80% train and 20% test, and a threshold was set at 0.2 to convert the anomaly scores into binary labels (1 for anomalies, 0 for non-anomalies). We used the same dataset for anomaly detection using Isolation Forest which identifies instances that deviate from the norm through isolation. The model was trained on unlabeled data, by skipping the existing labels in

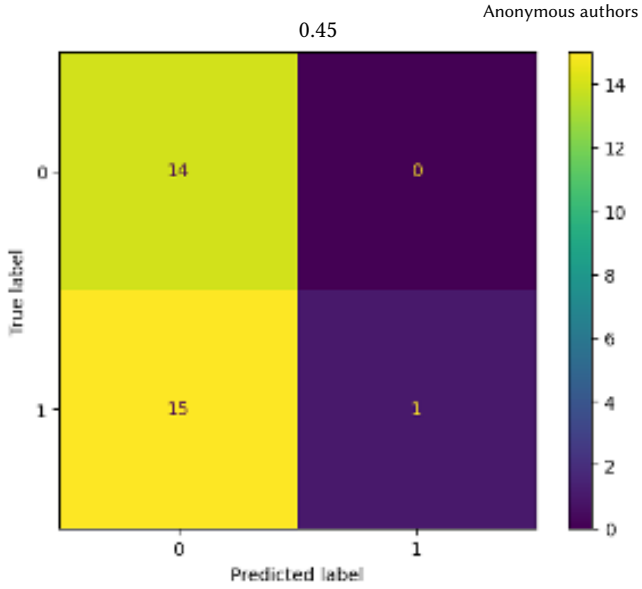
the original data. Figures 6 and 7 show the results by comparing the predicted anomalies returned from the XGBOD and Isolation Forest model.

The recall for the fraudulent class was high, meaning the model detected the most fraudulent cases in the test set. However, the precision was lower, indicating that many normal transactions were misclassified as fraud. This is typical in imbalanced datasets, where maximizing recall often comes at the cost of precision. For illustration, we did not conduct extensive threshold tuning or employ advanced techniques like ROC curve analysis or Precision-Recall curve optimization, which could improve the model's precision and recall balance.



	0	1	accuracy	macro avg	weighted avg
precision	0.888889	0.909091	0.9	0.89899	0.9
recall	0.888889	0.909091	0.9	0.89899	0.9
f1-score	0.888889	0.909091	0.9	0.89899	0.9
support	9.000000	11.000000	0.9	20.00000	20.0

Figure 5: Supervised ML Results Using XGBOD for Anomaly Detection on Fraud Dataset



	0	1	accuracy	macro avg	weighted avg
precision	0.482759	1.000000	0.5	0.741379	0.758621
recall	1.000000	0.062500	0.5	0.531250	0.500000
f1-score	0.651163	0.117647	0.5	0.384405	0.366621
support	14.000000	16.000000	0.5	30.000000	30.000000

Figure 6: Unsupervised ML Results Using Isolation Forest for Anomaly Detection on Fraud Dataset

6 Conclusion

DataOps has emerged as a pivotal field within AI in finance and investment banking, offering substantial benefits such as improved anomaly detection, automation of repetitive tasks, and enhanced decision-making through data-driven insights. This paper has proposed two primary methods for data quality management: data profiling using rule-based QC and anomaly detection using ML-based QC, demonstrated through various case studies. The application of the Advanced QC framework on an internal sample portfolio management dataset effectively identified outliers, with email reports highlighting significant deviations. The implementation of supervised and unsupervised machine learning algorithms for outlier detection further validated the efficacy of these methods across different datasets. In conclusion, by regularly analyzing these aspects, organizations can proactively address data quality issues, preventing potential downstream impacts on analyses, models, or business decisions in the financial industry. These practices not only streamline operations and ensure regulatory compliance but also provide a robust foundation for informed decision-making and effective risk management. By implementing comprehensive data quality control measures, financial institutions can enhance the accuracy and reliability of their data, leading to more trustworthy insights and better business outcomes.

Table 1: Overview of basic and advanced QC aggregated and individual columnar checks

Check Name	Data QC Rules	Examples
Aggregate-count	File count should be within a certain specified threshold	Count of rows present in the file should be between 60000 and 80000
Aggregate-null-diff-percentage-change	Calculate the null value percentage difference between yesterday's and today's data, in a particular column, should be within a specified threshold	Percentage change between today and yesterday null values of "volume" column should be below 10%
Aggregate-mean-diff-percentage-change	Calculate the mean value percentage difference between yesterday and today's data in a particular column, should be within a specified threshold	Percentage change between today and yesterday's mean values of "volume" column should be below 10%
Aggregate-correlation-change	Correlation for a given column between yesterday and today's data should be in specified threshold	Correlation between today and yesterday's "volume" column should be above 0.9
Aggregate-col-null-values	Percentage of null values present in a column should be within a specified threshold	Percentage of null values in "volume" column should be below 15%
Aggregate-common-cusips-change	Percentage of common cusips that have changed values day over day for given column	Percentage change in the value of "rating" column over common "cusips", between today's and yesterday's data should be below 5%
Aggregate-compare-col-names	Comparing columns names for yesterday's and today's data should be same across days	Column names should be constant comparing today's and yesterday's data
Security-null-value	Check for null values in a column which should not be present	"Date" column should not have any null values
Security-dup-value	Duplicates values should not be present in a column	"cusip" column should only contain unique values
Security-col-value	Values present in a column should be present in specified threshold	Each "cusip" amount issued should be between certain values
Security-col-ratio-check	Ratio of two columns should be less than the specified threshold	Ratio of "volume" by "amount issue" column should be less than 10%
Security-multiday-check	Check if the value of a column for two consecutive days is constant	Sum of a column at "cusip" level should not be 0
Security-missing-file-check	Check that the required file over which QC checks should be conducted is present and has required data	File for the current date should not be empty or unavailable, otherwise send an alert that file is missing for that day/week/month

Table 2: Implementations of various ML models in financial industry to resolve data quality issues

Check Name	Data QC Rules	Examples
Supervised	Extreme Boosting Based Outlier Detection (XGBOD) [40]	Using labelled data points, distinguish between normal data points and anomalous points with precision
Supervised	Random Forest Model	Using historical labelled data points, the model identifies exceptions as either true positives or false positives on new datasets
Supervised	Multiple Linear Regression Analysis	Forecast returns of securities based on different factors by calculating the volatility of returns relative to overall market price
Unsupervised	Isolation Forests	Outlier Detection on the unlabelled dataset by efficiently isolating anomalies using a tree structure on high dimension data
Unsupervised	Kullback-Leibler Divergence (KL Divergence) [21]	Capture drift and skewness on data points to assess the divergence between the distribution of incoming data and the distribution of the training data
Unsupervised	Principal Component Analysis (PCA) [19]	Used jointly with clustering methods for transaction monitoring in fraud detection
Semi-supervised	Generative Adversarial Networks (GANs) [14]	Detect anomalous data points on financial data by identifying if a given datapoint is real or has been generated synthetically
Reinforcement Learning	Gated Deep Q-networks [Van Hasselt 2016]	Automatically detecting and resolving data inconsistencies on vast amounts of financial data available

References

[1] 2024. The Crucial Role of Data Quality in Financial Services. https://www.castordoc.com. Accessed: 2024-10-25.

[2] 2024. The Impact of Poor Data Quality: Risks, Challenges, and Solutions. https://www.dataladder.com. Accessed: 2024-10-25.

[3] Inc. Amazon Web Services. 2023. Creating High-Quality Machine Learning Models for Financial Services Using Amazon SageMaker Autopilot. https://aws.amazon.com/blogs/machine-learning/creating-high-quality-machine-learning-models-for-financial-services-using-amazon-sagemaker-autopilot. Accessed: 2024-10-25.

[4] T. O. Ayodele. 2010. Types of Machine Learning Algorithms. In *New Advances in Machine Learning*, Yagang Zhang (Ed.). IntechOpen.

[5] A. Bakumenko and A. Elragal. 2022. Detecting Anomalies in Financial Data Using Machine Learning Algorithms. *Systems* 10, 5 (2022), 130.

[6] E. Caveness and P. S. GC. 2020. Ethical Issues in Data Science: From Privacy to Bias. In *International Conference on Data Science and Engineering*. Springer, 195–204.

[7] X. Chu, I. F. Ilyas, S. Krishnan, and J. Wang. 2016. Data Cleaning: Overview and Emerging Challenges. In *Proceedings of the 2016 International Conference on Management of Data (SIGMOD '16)* (New York, NY, USA). Association for Computing Machinery, 2201–2206. https://doi.org/10.1145/2882903.2912574

[8] Zha D., Bhat P. Z., Lai K., Yang F., Zhimeng J., Zhong S., and Hu X. 2018. Data-centric Artificial Intelligence: A Survey. https://arxiv.org/abs/2303.10158. Accessed: 2024-10-25.

[9] DenzilG. 2023. Corporate Bonds Indices [Dataset]. https://www.kaggle.com/datasets/denzilg/corporate-bonds-indices. Accessed: 2024-10-25.

[10] M. Dreves, G. Huang, Z. Peng, N. Polyzotis, E. Rosen, and P. S. GC. 2021. Validating Data and Models in Continuous ML Pipelines. *IEEE Data Eng. Bull.* 44, 1 (2021), 42–50. Retrieved from Google Scholar.

[11] T. Emmanuel, T. Maupong, D. Mpoeleng, T. Semong, B. Mphago, and O. Tabona. 2021. A survey on missing data in machine learning. *Journal of Big Data* 8 (2021), 1–37. https://link.springer.com/article/10.1186/s40537-021-00516-9

[12] S. Han, X. Hu, H. Huang, M. Jiang, and Z. Yue. 2022. ADBench: Anomaly Detection Benchmark. arXiv:2206.09426 https://arxiv.org/abs/2206.09426

[13] N. J. Horton and K. P. Kleinman. 2007. Much Ado About Nothing: A Comparison of Missing Data Methods and Software to Fit Incomplete Data Regression Models. *The American Statistician* 61, 1 (2007). http://www.math.smith.edu/~nhorton/muchado.pdf

[14] J. Ian, P. Jean, M. Mehdi, X. Bing, and W. David. 2014. Generative Adversarial Networks. *arXiv* (2014). https://arxiv.org/abs/1406.2661

[15] Marco Iansiti and Karim R. Lakhani. 2020. *Competing in the Age of AI: Strategy and Leadership When Algorithms and Networks Run the World.* Harvard Business Review Press.

[16] DataOps Institute. 2020. DataOps: A new approach to data management. DataOps Institute.

[17] M. Janssen. 2020. *Data Quality: A Key to Business Success.* Springer.

[18] M. Jiang, G. An, Y. Zhang, J. Wu, Liu W., and Y. Ding. 202. ADBench: Anomaly Detection Benchmark. https://github.com/Minqi824/ADBench.

[19] I. Jolliffe. 2011. Principal Component Analysis. In *International Encyclopedia of Statistical Science*, M. Lovric (Ed.). Springer. https://doi.org/10.1007/978-3-642-04898-2_455

[20] Ramin Karim, Diego Galar, and Uday Kumar. 2023. *AI Factory: Theories, Applications and Case Studies. ICT in Asset Management.* CRC Press, Taylor & Francis Group.

[21] J. F. Kurian and M. Allali. 2024. Detecting drifts in data streams using Kullback-Leibler (KL) divergence measure for data engineering applications. *Journal of Data, Information, and Management* 6 (2024), 207–216. https://doi.org/10.1007/s42488-024-00119-y

[22] R. Mahanti. 2019. *Data quality: Dimensions, Measurement, Strategy, Management, and Governance.* Quality Press.

[23] S. Moore. 2018. How to Stop Data Quality Undermining Your Business. https://www.gartner.com/smarterwithgartner/how-to-stop-data-quality-undermining-your-business. Accessed: 2024-10-25.

[24] A. B. Nassif, M. A. Talib, Q. Nasir, and F. M. Dakalbab. 2021. Machine learning for anomaly detection: A systematic review. *IEEE Access* 9 (2021), 78658–78700. https://ieeexplore.ieee.org/abstract/document/9439459

[25] MIT News. 2023. MIT Researchers Introduce GenSQL for Advanced Database Models. https://news.mit.edu/2023/gensql-generative-ai-databases. Accessed: 2024-10-25.

[26] J. Nonnenmacher and J. M. Gómez. 2021. Unsupervised anomaly detection for internal auditing: Literature review and research agenda. *International Journal of Digital Accounting Research* 21 (2021).

[27] National Institute of Standards and Technology (NIST). [n. d.]. About NIST. https://www.nist.gov/about-nist. Accessed: 2024-10-25.

[28] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and É. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830. https://doi.org/10.5555/1953048.2078195

[29] L. Petersen. 2023. Data Management in Financial Services: Practices and Implications. *Financial Analysis Journal* 78, 5 (2023), 34–45.

[30] N. Polyzotis, M. Zinkevich, S. Roy, E. Breck, and S. Whang. 2019. Data validation for machine learning. *Proceedings of Machine Learning and Systems* 1 (2019), 334–347. Retrieved from Google Scholar.

[31] T. C. Redman. 2018. *Data Driven: Creating a Data Culture.* Harvard Business Review Press.

[32] Lange D. Schmidt P. Celikel M. Biessmann Felix. Grafberger A. Schelter, S. 2018. Automating large-scale data quality verification. *Proc. VLDB Endow.* 11, 12 (2018), 1781–1794. https://doi.org/10.14778/3229863.3229867

[33] Astera Software. 2023. Data Governance in Financial Services: A Complete Analysis. https://www.astera.com/type/blog/data-governance-financial-services. Accessed: 2024-10-25.

[34] IBM Industry Solutions. 2023. Real-time Data Streaming in Financial Markets. https://www.ibm.com/blog/real-time-data-streaming-financial-markets. Accessed: 2024-10-25.

[35] Federal Reserve System. 2011. Supervisory Guidance on Model Risk Management (SR 11-7). https://www.federalreserve.gov/supervisionreg/srletters/sr1107.htm. Accessed: 2024-10-25.

[36] TensorFlow. [n. d.]. TensorFlow Data Validation. https://www.tensorflow.org/tfx/guide/tfdv. Accessed: 2024-10-25.

[37] V. M. Wiel. 2024. New AI Survey: Poor Data Quality Leads to $406 Million in Losses. https://www.fivetran.com/blog/new-ai-survey-poor-data-quality-leads-to-406-million-in-losses. Accessed: 2024-10-25.

[38] V. Yandrapalli. 2024. AI-Powered Data Governance: A Cutting-Edge Method for Ensuring Data Quality for Machine Learning Applications. In *2024 Second International Conference on Emerging Trends in Information Technology and Engineering (ICETITE)* (Vellore, India). IEEE, 1–6. https://ieeexplore.ieee.org/abstract/document/10493601

[39] Y. Zhang, J. Jiang, and X. Jin. 2020. Anomaly Detection in Data Streams Using Machine Learning Models. *IEEE Access* 8 (2020), 133926–133937. https://doi.org/10.1109/ACCESS.2020.3010627

[40] Y. Zhao and M. K. Hryniewicki. 2018. XGBOD: Improving supervised outlier detection with unsupervised representation learning. In *2018 International Joint Conference on Neural Networks (IJCNN)* (Rio de Janeiro, Brazil). IEEE. https://doi.org/10.1109/IJCNN.2018.8489483

[41] Y. Zhao, Z. Nasrullah, and Z. Li. 2019. PyOD: A Python Toolbox for Scalable Outlier Detection. *Journal of Machine Learning Research (JMLR)* 20, 96 (2019), 1–7.

Received 21 October 2024; accepted 24 October 2024