Predicting data quality of event-based container trackers

Daniël Hogendoorn

Delft University of Technology

Predicting data quality of event-based container trackers

by

Daniël Hogendoorn

Master Thesis

in partial fulfilment of the requirements for the degree of

Master of Science
in Mechanical Engineering

at the Department Maritime and Transport Technology of Faculty Mechanical Engineering of Delft University of Technology
to be defended publicly on Friday August 29th, 2025 at 3:30 PM

Student number:	5336252	
MSc track:	Multi-Machine Engineering	
Report number:	2025.MME.9097	
Supervisor:	dr. F. Schulte	
Thesis committee:	Dr. ir. Y. Pang	TU Delft committee member, ME
	Dr. N. Yorke-Smith,	TU Delft, committee member, EEMCS
	Dr. ir. B. van Riessen	company supervisor, Poort8
Date:	August 22nd, 2025	

Cover: Container carrier waiting to take off, by Nathan Cima

Preface

This thesis marks the end of a wonderful time I have had at TU Delft. First, during my Bachelor in Mechanical Engineering, I discovered that mechanical design did not spark my greatest interest, while the systems side truly captured my attention. Therefore, I enrolled in the Master track Multi-Machine Engineering, and what a blast that has been. Learning so much more about logistics and systems of systems was something I could not have imagined to be this enjoyable. Now, these two years are behind me. However, I have just started a job at Poort8, where I also conducted my graduation research. In my role there, I will continue to be involved in container logistics and, as such, keep learning about logistics.

This graduation research could not have been completed without the support of some very important people. Usually, I am not one to really speak up about this, but for this milestone in my life, I will make an exception:

First of all, the members of my graduation committee: Frederik, Neil, and Yusong. Frederik, thank you for taking time every week to meet with the graduate students. Thank you for pointing me in the right direction, providing useful starting points whenever I tried a new approach, and connecting me to the right people. Neil, although we had never met before the start of this graduation research, thank you for carefully reading my drafts and asking the critical questions that helped me see where improvements were needed. Yusong, although you only joined this research towards the end, thank you for being willing to participate in the committee.

Next, this thesis could not have been written without the help of the people at Poort8. Cunes, thank you for your practical approach to problems and for giving me much-needed programming advice. Amy, thank you for always being available to answer any question I had, even the silly ones. Also, Gert, thank you for helping me whenever I did not know how to continue, and for all the small talk on Fridays. Lastly, Bart, you were amazing at quickly pointing out when I went off track in my research, and showing me how I could get back on course. Thank you for that.

Finally, there are some very important people close to me who witnessed both my struggles in the beginning and my eventual success. Since they are all Dutch-speaking, I will continue in Dutch from here.

Lastly, there are some very important people that are very close to me and have seen the process of me struggling with my research at the start, and eventually succeeding in this research. Since they are all dutch-speaking, I'll continue in Dutch from here on.

Lieve papa en mama, bedankt dat jullie altijd voor me klaarstonden in de afgelopen vijf jaar. Het was soms niet altijd even makkelijk, maar ondanks dat stond er altijd een maaltijd voor mij klaar. Jullie zijn altijd geïnteresseerd geweest in mijn opleiding, ook al snapten jullie er soms niet veel meer van. Dan nog mijn aanstaande schoonouders, bedankt voor al jullie goede zorgen door de tijd heen. Dank voor het geruststellen als ik soms zelf dacht dat mijn afstudeeronderzoek niets ging worden.

Het laatste woord is voor jou, Jorine. Dankjewel dat je er altijd voor me was. Het was heel leuk om jou soms stress te zien hebben voor tentamens, en vervolgens zelf stress te krijgen over mijn eigen tentamens die drie weken later waren. Jammer dat mijn geruststellende woorden van eerder niet op mij werkten :). Niettemin, we zijn nu allebei klaar met onze opleiding en gaan allebei beginnen aan onze professionele carrière. Ik kijk ernaar uit om straks samen met jou de deur uit te lopen, richting het station of op de fiets naar Rotterdam. Je bent fantastisch.

Soli Deo Gloria

Daniël Hogendoorn
Zwijndrecht, August 2025

Abstract

Reliable container-tracking depends on the quality of estimated time-of-arrival (ETA) data, yet existing logistics platforms offer little guidance on how trustworthy those timestamps really are. This thesis proposes a *fit-for-use* data-quality (DQ) framework for Digital Container Shipping Association (DCSA)–compliant event logs that flags ETA records likely to deviate from actual time of arrival (ATA) by more than one calendar day.

Event logs from $\sim$90 k transport legs were preprocessed into records capturing origin-destination pair, carrier, publisher type, and timing information. Four supervised models, namely Linear Regression (LR), Random Forest, XGBoost, and a Neural Network, were trained to predict leg duration. A prediction that placed ATA > 1 day from the published ETA labeled that record *low-quality*. Model outputs were evaluated with a precision-oriented F_β-score, where a false alarm is 50 times more costly than a missed detection ($\beta \approx 0.141$).

The simplest model prevailed: standard LR achieved the highest overall $\mathrm{F}_{0.141}$-score (68.5 %), balancing few false positives with robust recall, while more-complex tree-based and neural models produced excessive false alarms. When the analysis was narrowed to early-stage ETAs published by carriers (arguably the least reliable yet most operationally valuable subset) LR's score rose to 72.0 %. These findings highlight that careful feature engineering and data curation outweigh algorithmic complexity for this task.

The study delivers the first systematic, event-data-only method to quantify DQ in container tracking, enabling near-real-time plausibility checks without AIS feeds. Limitations include a three-month observation window and absence of exogenous factors such as weather or port congestion. Future work should extend the temporal scope, integrate AIS-derived and environmental features, and explore meta-learning techniques to adapt to disruptions. It could also use process-mining to uncover anomalous event sequences to take a different approach in dataquality assessment within container-eventlogs.

By demonstrating that a transparent LR baseline can reliably surface dubious ETAs, the thesis provides a practical blueprint for logistics platforms seeking to bolster trust in their tracking data and to prioritise corrective action where it matters most.

Summary

Containerized shipping underpins nearly 90% of international trade, and reliable container-tracking is essential for the timely and cost-efficient functioning of global logistics networks. Yet, despite increasing adoption of the Digital Container Shipping Association (DCSA) Track & Trace (T&T) standard, data quality (DQ) within event-based container trackers remains inconsistent. Stakeholders face the recurring problem that estimated times of arrival (ETAs) are often unreliable, undermining trust in tracking platforms and complicating operational planning.

This thesis investigates how the quality of container-tracking data in a DCSA-compliant environment can be systematically assessed. It proposes a predictive framework that classifies event data as high- or low-quality based on whether ETAs deviate from actual times of arrival (ATAs) by more than one day. The study thereby contributes to a practical definition of “fit-for-use” data in logistics, aligning with operational needs for early detection of unreliable records.

The research adopts a three-phase methodology. First, a literature review identifies the limitations of existing DQ assessment methods, which largely focus on individual dimensions such as accuracy, completeness, or timeliness. While valuable, these metrics do not suffice to judge whether a container journey is represented plausibly as a whole. Second, the structure of DCSA event logs is analyzed, including the segmentation of container journeys into transport legs. This step demonstrates how event-level data can be transformed into a tabular format suitable for machine learning. Finally, supervised models, Linear Regression (LR), Random Forest (RF), Extreme Gradient Boosting (XGBoost), and Neural Networks (NN), are trained on approximately 90,000 transport legs collected via the HeyWim Container Tracking system, covering April to July 2025.

Model evaluation shows that simplicity outperforms complexity. Standard linear regression achieved the highest $F_{0.141}$ score (68.5%), reflecting a balance between correctly identifying low-quality ETAs and avoiding false alarms. This cost-sensitive metric assigns a false positive fifty times the penalty of a false negative, reflecting industry priorities: spurious alerts are deemed more disruptive than missed detections. By contrast, tree-based and neural models frequently overfitted, generating excessive false alarms and proving less suitable for production use. Importantly, performance improved further when focusing on carrier-provided ETAs published at least one week before arrival, an especially unreliable yet operationally significant subset. In this restricted evaluation, linear regression reached a precision-oriented score of 72.0%, demonstrating its robustness in detecting the most critical quality issues.

The findings underscore two main insights. First, systematic preprocessing and domain-specific feature engineering are more decisive for predictive performance than algorithmic complexity. Second, the proposed framework shows that event-data-only approache, without reliance on AIS trajectories or other external feeds, can already provide actionable insights into data quality. This positions the method as a viable near-real-time plausibility check for logistics platforms.

The study also highlights several limitations. The dataset spans only four months and does not incorporate exogenous factors such as port congestion, weather, or geopolitical disruptions. Sparse coverage of certain origin–destination pairs further constrained model generalizability. Moreover, the “fit-for-use” criterion was researcher-defined rather than validated through user studies, which may overlook stakeholder-specific tolerances. Future work should therefore expand the temporal scope, integrate AIS and environmental data, and explore meta-learning and process-mining techniques to adapt to disruptions and identify anomalous event sequences.

In conclusion, this thesis presents the first systematic, predictive approach to assessing the quality of DCSA-compliant container event data. By demonstrating that a transparent linear regression baseline can reliably flag dubious ETAs, it offers a practical and scalable blueprint for logistics providers seeking to improve the trustworthiness of their tracking platforms. Beyond its direct contributions, the research opens pathways for hybrid models, user-driven definitions of “fitness,” and broader integration of contextual data sources. In doing so, it strengthens the foundation for data-driven, reliable, and scalable container-tracking solutions.

Samenvatting

Containervervoer is verantwoordelijk voor bijna 90% van de wereldhandel, en betrouwbare containervolgsystemen zijn cruciaal voor een tijdige en kostenefficiënte werking van mondiale logistieke netwerken. Ondanks de toenemende adoptie van de Digital Container Shipping Association (DCSA) Track & Trace (T&T) standaard blijft de datakwaliteit (DQ) binnen event-gebaseerde containervolgsystemen echter inconsistent. Belanghebbenden worden regelmatig geconfronteerd met het probleem dat geschatte aankomsttijden (ETAs) onbetrouwbaar zijn, waardoor het vertrouwen in volgsystemen afneemt en operationele planning wordt bemoeilijkt.

Dit onderzoek richt zich op de vraag hoe de kwaliteit van containervolgsystemen in een DCSA-conforme omgeving systematisch kan worden beoordeeld. Het introduceert een voorspellend raamwerk dat eventdata classificeert als hoog- of laagwaardig op basis van de afwijking tussen ETA en de werkelijke aankomsttijd (ATA). Wanneer deze afwijking groter is dan één dag, wordt de betreffende gebeurtenis als laagwaardig aangemerkt. Daarmee draagt dit onderzoek bij aan een praktische invulling van het begrip “fit-for-use” data in de logistiek, afgestemd op de operationele behoefte om onbetrouwbare records vroegtijdig te signaleren.

Het onderzoek volgt een drieledige methodologie. Ten eerste identificeert een literatuurstudie de beperkingen van bestaande methoden voor DQ-beoordeling, die grotendeels gericht zijn op afzonderlijke dimensies zoals nauwkeurigheid, volledigheid of actualiteit. Hoewel waardevol, schieten deze metrics tekort om te bepalen of een complete containerreis op plausibele wijze wordt weergegeven. Ten tweede wordt de structuur van DCSA-eventlogs geanalyseerd, inclusief de segmentatie van containerreizen in afzonderlijke transportlegs. Hiermee wordt aangetoond hoe eventdata kan worden omgezet in een tabelvorm die geschikt is voor machine learning. Ten slotte zijn vier supervisie-algoritmen, Lineaire Regressie (LR), Random Forest (RF), Extreme Gradient Boosting (XGBoost) en Neurale Netwerken (NN), getraind op circa 90.000 transportlegs, verzameld via het HeyWim Container Trackingsysteem over de periode april tot en met juli 2025.

De modelresultaten laten zien dat eenvoud beter presteert dan complexiteit. Standaard lineaire regressie behaalde de hoogste $F_{0.141}$-score (68,5%), waarmee een evenwicht werd bereikt tussen het correct identificeren van laagwaardige ETAs en het vermijden van valse alarmen. Deze kostenbewuste maatstaf kent aan een vals positief vijftig keer zoveel gewicht toe als aan een vals negatief, in lijn met industriële prioriteiten waarin foutieve waarschuwingen disruptiever worden geacht dan gemiste detecties. Meer complexe modellen, zoals boomgebaseerde methoden en neurale netwerken, bleken vaak te overfitten en genereerden te veel foutieve meldingen, waardoor zij minder geschikt zijn voor operationeel gebruik. Opvallend is dat de prestaties verder verbeterden wanneer uitsluitend werd gekeken naar door rederijen verstrekte ETAs die minimaal een week voor aankomst waren gepubliceerd. Binnen deze subset, die vaak het meest onbetrouwbaar maar ook operationeel het meest waardevol is, behaalde lineaire regressie een precisiegerichte score van 72,0%, waarmee de robuustheid van dit model in het detecteren van kritieke kwaliteitsproblemen werd bevestigd.

De resultaten onderstrepen twee hoofdpunten. Ten eerste zijn systematische voorbewerking en domeinspecifieke feature engineering bepalender voor de voorspellende prestaties dan algoritmische complexiteit. Ten tweede toont het voorgestelde raamwerk aan dat benaderingen die uitsluitend op eventdata zijn gebaseerd, zonder gebruik van AIS-data of andere externe bronnen, reeds bruikbare inzichten kunnen leveren in datakwaliteit. Dit positioneert de methode als een levensvatbare controle voor plausibiliteit in (bijna) real-time voor logistieke platforms.

Het onderzoek wijst tevens op enkele beperkingen. De dataset bestrijkt slechts vier maanden en houdt geen rekening met externe factoren zoals havencongestie, weersomstandigheden of geopolitieke verstoringen. Daarnaast werd de generaliseerbaarheid beperkt door de schaarse aanwezigheid van bepaalde herkomst–bestemmingsparen. Bovendien werd het criterium “fit-for-use” door de onderzoeker gedefinieerd, zonder validatie bij gebruikers, waardoor mogelijk niet alle praktijkrelevante toler-

anties zijn meegenomen. Aanbevolen vervolgonderzoek zou daarom de temporele scope moeten uitbreiden, AIS- en omgevingsdata integreren, en meta-learning en process mining verkennen om zowel verstoringen als afwijkende eventvolgorden te detecteren.

Samenvattend presenteert dit onderzoek de eerste systematische, voorspellende benadering voor het beoordelen van de kwaliteit van DCSA-conforme containereventdata. Het toont aan dat een transparant lineair regressiemodel betrouwbaar twijfelachtige ETAs kan signaleren en biedt daarmee een praktisch en schaalbaar kader voor logistieke dienstverleners die het vertrouwen in hun volgsystemen willen vergroten. Naast de directe bijdrage opent dit onderzoek wegen naar hybride modellen, gebruikergedefinieerde definities van “fitness” en bredere integratie van contextuele databronnen. Daarmee verstevigt het de basis voor datagedreven, betrouwbare en schaalbare oplossingen voor containertracking.

Nomenclature

Abbreviations

Abbreviation	Definition
AIS	Automatic Identification System
ATA	Actual Time of Arrival
DQ	data quality
ETA	Expected Time of Arrival
FN	False Negative
FP	False Positive
LLM	Large Language Model
LR	Linear Regression
ML	Machine Learning
OD	origin-destination
RF	Random Forest
TN	True Negative
TP	True Positive

Contents

List of Figures

List of Tables

1 Introduction

It has been estimated that about 90% of the world's trade is transported in cargo containers (Camossi, Dimitrova, and Tsois, 2012), making containerized shipping a cornerstone of global commerce. Container tracking plays a crucial role in ensuring that goods move smoothly across international logistics networks. Shippers, carriers, and end-users rely on timely, accurate information to plan routes efficiently and avoid costly delays. Yet, as the velocity and volume of shipments increase, so do the demands placed on the underlying data infrastructure. Small errors in container location, transit milestones, or estimated arrival times can compound quickly, resulting in misplaced goods, scheduling conflicts, and higher operating costs [2]. These challenges underscore the necessity of robust data quality (DQ) within container tracking systems.

Compounding the problem is the diversity of data sources. Major concerns for container tracking include missing or inconsistent event records and inaccurate timestamps. Data can originate from shipping lines' enterprise systems, port authorities' databases, or third-party logistics providers, each with different formats and reliability standards. This complexity amplifies the risk of poor DQ and compromises the ability of stakeholders to trust the system's outputs [3].

A recent development shaping the digital transformation of container logistics is the founding of the Digital Container Shipping Association (DCSA). Some of the major shipping companies (i.e. MSC, Maersk, CMA CGM, Hapag-Lloyd, ONE, Evergreen, Yang Ming, HMM, and ZIM) formed the DCSA to establish shared information technology standards across the industry. By promoting a unified digital language, the DCSA aims to improve data interoperability and enable seamless communication between everyone within the industry [4]. One of the standards the DCSA has developed is the DCSA Track & Trace (T&T) standard. This standard is an API specification that standardizes event definitions (i.e. timestamped records of specific logistics milestones), which allows shippers, carriers, and other supply-chain partners to exchange container-movement information in a consistent way. With growing adoption, this standard is helping to align a wide range of tracking systems and software providers around consistent semantics and data structures.

However, standardization alone does not guarantee the quality of the data being exchanged. Even DCSA-compliant platforms may propagate incomplete, delayed, or erroneous information. This thesis addresses that gap by exploring a preliminary approach to monitoring DQ in a DCSA-compliant container tracking system. The resulting findings aim to identify the possibilities of DQ monitoring in tracking platforms that adhere to DCSA standards. Ultimately, this research contributes to ensuring that container event data are not only timely but also trustworthy and fit for operational use.

1.1. Research questions

The purpose of this thesis is to contribute to an understanding of how DQ can be assessed in container-tracking systems to foster accurate information for global logistics operations. Specific objectives are:

- To evaluate the applicability of existing DQ assessment methods to event-based data.

- To develop new DQ assessment methods that can detect whether or not a container journey is ”fit-for-use”.

To achieve these objectives, the central research question in this thesis is:

How can data quality in a DCSA-based container tracker be systematically assessed to support accurate and scalable container tracking?

To answer this question, the following sub-questions are posed:

1. *What data quality assessment methods are suitable for evaluating event-based tracking data?*
2. *How can the analysis of individual transport-legs within container journeys be used to systematically assess data quality?*

Study limitations

While this research systematically assesses and seeks to improve DQ in DCSA–compliant container tracking, several constraints temper the generalisability of its findings.

1. **Quantitative focus without user validation.** ‘Fitness for use’ is widely recognized as a central concept in DQ research. However, it is assessed solely with researcher-defined metrics; no interviews or surveys were conducted to capture stakeholder perceptions of error criticality. Consequently, some defects flagged by the model may be operationally benign and vice versa.
2. **Restricted temporal window.** The evaluation is performed on a historic dataset covering the months April, May, June, and July 2025, because ground-truth data were available for that period. This limited time window means the results should be viewed as indicative rather than definitive and should inspire further research across additional periods.
3. **No real-time performance benchmarking.** Latency, throughput, and computation cost metrics are not measured under live streaming loads, leaving open the question of readiness for production deployment.
4. **Dependence on DCSA Track & Trace version 3.0.** All parsing logic and validation rules target the current schema. Future DCSA releases could introduce slightly changed fields of field definitions, necessitating minor re-engineering.

1.2. Methodology

To arrive at a conclusive answer to the research question, the study adopts the following three-phase methodology:

1. **Literature Review on DQ Assessment Methods**
   A systematic literature review will identify established methods for assessing DQ. The review will provide a rationale for selecting or adapting metrics and evaluation techniques suitable for container-tracking data.
2. **Analysis of the Role and Structure of DCSA Event Data**
   The study then establishes a detailed understanding of the DCSA Track & Trace schema and its implementation, cataloguing attributes, dependencies, and use-cases critical for quality assessment. This structural analysis ensures that subsequent techniques remain applicable across DCSA-based container-tracking systems.
3. **Transport-Leg-Level DQ Assessment**
   Because existing methods do not fully align with the requirements of a container tracker, the quality of event data will be assessed at the transport-leg level, defined as the continuous movement of a container on a single mode of transport equipment. Machine learning (ML) algorithms (linear regression, extreme gradient boosting, random forests, and neural networks) will predict arrival times. When the predicted arrival time differs substantially from the ETA recorded in the data, the corresponding data point will be flagged as low-quality.

Each step of this methodology is described in more detail in the next chapters. Specifically, Chapter 2 contains a literature review of DQ assessment methods. Chapter 3 explores container-event data in greater depth. Chapter 4 then explains the machine learning approach and presents the results. Chapter 5 discusses limitations and proposes avenues for further research. Finally, Chapter 6 summarises the principal conclusions of this study.

2

Literature Review

High-quality data is essential for enabling information-driven decision-making in Logistics 4.0, yet many organizations still face persistent data quality issues (Xie, Sun, and Zhao, 2025).

Most scholars agree that DQ can be assessed through a range of attributes, including completeness, accuracy, timeliness, validity, periodicity, relevance, reliability, and precision (Chen et al., 2014). Due to the abundance of such attributes, many researchers have proposed categorizing them into broader dimensions to facilitate evaluation (Hazen et al., 2014).

Wang and Strong (1996) categorized 15 attributes in 4 categories; intrinsic, contextual,representational and accessibility. Askham et al. refined this classification in 2013 by reducing the list of 15 attributes to six core attributes: completeness, uniqueness, timeliness, validity, accuracy, and consistency. This reduction negated the need for categorization and represents a widely accepted approach (Xie, Sun, and Zhao, 2025).

However, there is considerable variation in both the terminology and the structure used to categorize DQ attributes. For instance, Hazen et al. (2014) employs the term *dimensions* to describe individual attributes, and categorizes these as either *intrinsic* (those native and objective to the data) or *contextual*, referring to attributes that depend on the circumstances in which the data is used.

Conversely, Cai and Zhu (2015) proposes a different categorization, introducing five categories: availability, usability, reliability, relevance, and presentation quality. In this framework, the term ”dimensions” is used to denote categories, while ”elements” correspond to individual attributes. So, whereas in Hazen et al. dimensions is used to describe attributes, here the term dimensions is used to describe categories.

These variations highlight the lack of consensus on both the structure and terminology of DQ categories and attributes, a challenge also noted by Chen et al. (2014).

Despite this lack of clarity, the attributes scholars mention most often are accuracy, timeliness, consistency, and completeness (Hazen et al. (2014) and Wang, Hulstijn, and Tan (2016), and Batini et al. (2009)). In the literature they are described as follows:

- **Accuracy** can be assessed by comparing values with external values that are known to be (or considered to be) correct (Hazen et al., 2014). This can be done by comparing the value to a trusted source of truth.
- **Completeness** is defined as the degree to which a given data collection includes data describing the corresponding set of real-world objects (Batini et al., 2009). It characterizes the number of missing values. A value can be missing for the following three reasons:
    - because it exists, but is not known;
    - because it does not exist;
    - because it is not known whether it exists.

- **Consistency** can be seen as whether or not the representation of the data value is the same in all cases (Ballou and Pazer, 1985). Batini et al. (2009) introduces the concept of intra-relation constraints and inter-relation constraints. Intra-relation consistency refers to the value range a certain data value can have. Inter-relation is whether a value is the same in different databases. Inter-relation consistency might seem to overlap with accuracy, however, accuracy is only about whether the data value corresponds to its real-world equivalent. Inter-relation consistency is about values matching each other in different databases, so it is comparing different values to each other.
- **Timeliness** refers to the degree to which data are up-to-date (Hazen et al., 2014). Data may become outdated if it is not updated when source information changes. In the case of container trackers, timeliness can be compromised if initially incomplete data is not refreshed once it becomes available from the source, or when the source itself is updated.

When looking further for articles on how to assess DQ, authors other than those mentioned above continue to evaluate DQ solely on single dimensions (Bergdahl et al. (2007), Verhulst (2016), and Bokrantz et al. (2017)).

Bergdahl et al. (2007) provided a comprehensive framework for statistics agencies.Verhulst (2016) also evaluated data as a sum of single dimensions, and Bokrantz et al. (2017) charted 33 simulation-data issues, mapped these issues to nine quality dimensions and revealed major gaps in accessibility, completeness, consistency.

Table 2.1: Analysis of articles on DQ assessment

Reference	Focus	Approach	Key Contribution	Strengths	Limitations
Xie et al. [5]	DQ in ML design/manufacturing	Systematic literature review	Taxonomy of concepts and methods	Broad and up-to-date overview	Domain-specific
Chen et al. [6]	DQ in health information systems	Review of assessment methods	Classification of DQ methods and dimensions	Clear framework; domain examples	Health-specific
Hazen et al. [7]	DQ for supply-chain analytics	SPC control charts	Links DQ with SPC	Integrates SCM & data-science views	Only conceptual
Wang et al. [8]	Consumer perspective on DQ	Two-stage survey & factor analysis	4-category DQ framework	Seminal, widely adopted	Single-domain survey study
Cai et al. [10]	Big-data DQ challenges	5 dimensions & feedback cycle	Two-layer indicators per dimension	Maps 4V to metrics	Only conceptual
Batini et al. [12]	DQ assessment & improvement methods	Systematic 5-perspective comparative survey	Classified phases, steps, and cost factors	Comprehensive cross-domain synthesis	Only conceptual
Bergdahl et al. [14]	Statistical quality reporting	Assessment on dimensions	Structured implementation roadmap	Comprehensive framework	Primarily for statistics agencies
Verhulst [15]	Event DQ	Assessment on dimensions	Software plugin	Systematic approach	Individual value analysis
Bokrantz et al. [16]	DQ in DES	Interviewing professionals	Provides improvement guidelines	Investigates data production	Focus on manufacturing
Camossi et al. [1]	Container-route anomaly detection	SVM on CSM	End-to-end pipeline	Scales to 300 k real trips	Limited features & unsupervised
Chen et al. [17]	DQ improvement	Spectral Decomposition & VAT Partitioning	Nonlinear cluster detection	Highly visual method	High computation cost
Feiter et al. [18]	Fault detection	Machine Learning	Shows quality Assessment with ML	Clear feature selection	Not used on logistic data

2.1. Assessment Methods Beyond Dimensional Metrics

A handful of works move beyond single-attribute metrics and propose genuinely novel concepts or holistic approaches. Camossi, Dimitrova, and Tsois (2012) evaluates Container Status Messages (CSMs) to identify anomalous container routes. This research was primarily intended to be used by customs, so customs agents had a better idea on what containers to check. However, this method can with some re-engineering also be used to detect low-quality data points in container journeys, as anomalous container routes can also be a sign of incorrect data.

Chen, Zhu, and Lee (2013) presents a visual-assessment-based data-partitioning workflow that judges whether a dataset is “model-ready” before prognostic modelling begins. The method projects high-dimensional sensor data into a low-dimensional spectral space, applies a VAT-style image to expose natural clusters, and then quantifies each cluster’s internal fitness and mutual separation; outliers are flagged via a minimum-spanning-tree disparity analysis. The same logic can potentially be transferred to DCSA container-event streams, where implausible clusters or isolated outliers likewise signal low-quality transport-leg records.

Next, Feiter, Strickland, and Garcia-Marti (2025) recast wind-vane stalling as a machine learning (ML) classification task rather than rating data against predefined quality dimensions. Using twenty years of Cabauw-tower observations, they benchmark five supervised algorithms and a one-class SVM, finding that K-Nearest Neighbours and Random Forest detect stalling episodes with 75% accuracy while sharply reducing false positives.

As Table 2.1 demonstrates, most prior work relies on value-level or dimension-based evaluation frameworks; only a handful of studies explore clustering or outlier detection, and none address DCSA-compliant container-events. This gap underscores the novelty and relevance of the method developed in this thesis. The approach employs ML models to predict the actual arrival time at each planned destination, and any journey whose predicted arrival deviates by more than one day from the ETA recorded in the DCSA data is flagged as low-quality. Crucially, the method operates without Automatic Identification System (AIS) data, an input on which all comparable studies still depend, as Jiang et al. (2025) found in a comprehensive literature review.

Another way of looking at DQ is evaluating whether it is ’*fit-for-use*’ (Wang and Strong, 1996), which means that DQ is determined by whether the user can do something useful with is. This is also emphasized by Galway and Hanks (1996), which states that DQ can only be meaningfully judged in the context of its application.

While the dimensions discussed above provide useful indicators for evaluating individual data values, they fall short of assessing whether the data, as a whole, forms a coherent and trustworthy representation of reality. This is particularly relevant in the context of container tracking systems, where users interpret data at the level of entire container journeys. Thus, it is proposed that the quality of a container journey should not be assessed as the sum of isolated value-level metrics, but instead as a holistic judgment of plausibility and internal consistency. To this author’s knowledge, no such method currently exists in the literature. Therefore, this thesis develops a novel DQ assessment technique tailored to event-based container tracking. This method is already introduced briefly in Section 1.2, where the methodology for this research project is discussed. Chapter 4 will go into greater depth on this approach.

To summarize:

- There is no consensus on what framework to use in order to structurally assess DQ.
- There are a few common denominators, such as assessment on data attributes. However, the question remains of what attributes to use.
- Therefore, there exists an enormous research gap that this thesis fills for the field of container logistics
- This thesis fills that gap by applying ML models to assess the quality of the ETAs provided in event data

3

Data Processing

International container transport is orchestrated by dozens of actors. Shipping lines, terminals, rail and barge operators, freight forwarders/ Each maintains its own information system and vocabulary. To make these disparate data streams interoperable, the Digital Container Shipping Association (DCSA) has published the Track & Trace (T&T) standard. A machine-readable schema that defines what constitutes a transport "event," which attributes must accompany it, and how those attributes should be encoded.

This chapter explains how the T&T event model is structured and why those design choices matter for downstream analytics. We begin by mapping the standard's three event families: Shipment, Equipment, and Transport, and clarify the four allowed status codes (Planned, Estimated, Actual, and Requested). Next, we show how individual events are stitched together into traces and how a collection of traces forms an eventlog, the core data asset used throughout the remainder of this work. Finally, we discuss practical considerations when building such a log from heterogeneous sources, and show the data processing steps (duplicate suppression, UN/LO-code cleansing, and the segmentation of container journeys into distinct transport legs) needed, laying the groundwork for the ML techniques developed in Chapter 4.

3.1. Events in the DCSA T&T standard

The DCSA Track & Trace (T&T) standard provides a structured framework for representing container transport events in a unified, interoperable format. An event can be any important step in the lifecycle of a container, such as it being loaded on a ship, or it being delivered to the customer. Understanding the event types defined in this standard is essential for segmenting container journeys and assessing DQ across transport legs.

3.1.1. Event Types

In the DCSA T&T standard, events are organized into three groups: Shipment events, Equipment events, and Transport events.

- **Shipment events** pertain to the shipment of an object itself, such as furniture, cars, pieces of clothing, etc. These types of events often reference associated documents, such as booking details or transport instructions. They focus on tracking the lifecycle of a shipment, including its booking, documentation, and completion.
- **Equipment events** are linked to specific pieces of equipment being tracked. In this research, the equipment events will always be linked to containers, although these types of events are not exclusive to containers, as they can also be used to track other pieces of equipment. In the case of containers, these events represent actions such as a container being loaded onto a vessel, arriving at a terminal, or being dropped off at the customer.
- **Transport events** are associated with specific transport calls and are connected to transport-

related actions (i.e., arrivals and departures). In the T&T standard, they are used to track occurrences of the movement of transport vehicles (not only land vehicles, but also watercraft, railed vehicles and aircraft), such as vessels and barges, at facilities like deep-sea terminals and inland terminals.

In summary, transport events track movements; equipment events focus on the handling of shipping tools like containers; and shipment events document the lifecycle of a shipment.

Each event may carry one of four statuses: Actual (`ACT`), Planned (`PLN`), Estimated (`EST`), or Requested (`REQ`). Actual represents an event that has already occurred, Planned refers to an event that is scheduled to happen in the future, Estimated indicates an event with a predicted occurrence time or status, and Requested denotes an event that has been requested to occur, often as part of a process or workflow. The `REQ` status can only be assigned to events of the Shipment type.

An overview of all events in the DCSA standard is provided in Appendix B.

3.1.2. Creating an Eventlog

All events belonging to one specific process instance is called a trace. In container tracking, a trace would be all the events for one container journey. An eventlog is a collection of all these traces.

For container logistics, creating an eventlog means that container data is gathered from various sources, transformed into the DCSA format (if not already compliant), and assembled into an eventlog that reflects each shipment's progress. Table 3.1 provides an illustrative trace for container ABCD1234567. Certain mandatory DCSA fields are omitted here for simplicity, and column names are not always corresponding to the names in the DCSA standard; the table's purpose is solely to demonstrate the eventlog structure. The event data used in this thesis is kindly provided by Poort8, as part of their container tracker product *HeyWim Container Tracking*. The data is collected during the months of April, May, June and July of 2025. The data was collected once every working day, with the exception of some national holidays that occurred on a working day. *HeyWim* already contains logic to filter out duplicate events from different sources, so to provide the most reliable data for each event. This makes the fleet mix skewed towards Poort8's customer base, and when collecting it via other sources, this duplication filtering needs to be manually created.

Table 3.1: Event Log of Container ABCD1234567

Event Date-Time	Status	Event	Location	UN/LO Code	Mode	ModeName	Publisher
2025-04-14T16:00:00	ACT	GTOT	JODHPUR	INJDH	TRUCK		Hapag
2025-04-16T15:28:00	ACT	ARRI	MUNDRA	INMUN	TRUCK		Hapag
2025-04-21T10:37:00	ACT	LOAD	MUNDRA	INMUN	VESSEL	Charleston Express	Hapag
2025-04-21-T21:18:00	ACT	DEPA	MUNDRA	INMUN	VESSEL	Charleston Express	Hapag
2025-05-28T15:00:00	PLN	ARRI	DDE (Delta)	NLRTM	VESSEL	Charleston Express	ECT
2025-05-30T14:30:00	PLN	DEPA	DDE (Delta)	NLRTM	VESSEL	Charleston Express	ECT
2025-05-30T22:00:00	PLN	DEPA	ROTTERDAM	NLRTM			Hapag
2025-06-02T22:00:00	PLN	ARRI	VENLO	NLVEN			Hapag

Table 3.1 shows the journey of a container from Jodhpur, India, to Venlo, the Netherlands The container

is transported on a truck to Mundra Port in India, where it is loaded on the Charleston Express. This vessel transports the container to Delta Terminal in Rotterdam, The Netherlands. From there, the container is further transported to Venlo. At the time of writing, the arrival event at Delta Terminal remains “Planned,” indicating the vessel is still en route.

It is also interesting to note there are multiple publishers of the data. ECT (the operator of the Delta terminal) is not providing information in DCSA, while Hapag (the shipping company Hapag-Lloyd) is one of the founders of DCSA and thus providing data according to the DCSA T&T standard. This has an impact on the data quality of the event log. Integrating a non-standard ECT feed requires a robust normalization and reconciliation layer. ECT’s proprietary status codes, local-time timestamps, and non-DCSA field names must be mapped into the standard schema, introducing a potential source of inconsistency if mappings are incomplete or drift over time. This shows the need for a thorough monitoring of data quality.

Although it may initially appear that the presence of multiple data sources warrants a comprehensive assessment of data quality across all dimensions (consistency in particular) this is not the primary intent. Rather, this example illustrates that while established data quality dimensions provide a useful analytical lens, they should neither be disregarded nor treated as a rigid framework. For instance, shifting focus from consistency to completeness highlights that the *Mode* field is empty for the segment between Rotterdam and Venlo. This is likely because the shipping company has not yet arranged or confirmed onward transport for the container. Given that the distance between these locations is approximately 200 kilometers, the container could plausibly be moved by truck, barge, or rail. In this case, the absence of a *Mode* value reflects operational uncertainty rather than a data quality failure per se, illustrating the limitations of relying exclusively on predefined dimensions for evaluation.

3.2. Transport-leg definition

A transport-leg is defined as being one container move on a single mode of transport. So if we take the event log from Table 3.1, there would be three legs:

- Jodhpur → Mundra on a truck
- Mundra → Rotterdam on the Charleston Express
- Rotterdam → Venlo on a transport vehicle yet to be determined

Identifying transport-legs within a container event log is a relatively straightforward task for a human observer. However, automating this process presents several challenges. As a first step, the event log must be sorted chronologically based on the *Event DateTime*. While iterating through the ordered events, one might consider using a heuristic whereby a change in transport mode indicates the beginning of a new leg. However, such a rule is insufficient. For instance, it is not uncommon for containers to be transshipped between vessels. If the corresponding discharge and load events are missing (whether due to incomplete reporting or data quality issues) the sequence of events looks like the container is on the same mode of transport, while in reality, the container has moved from one vessel to another. To mitigate this, an additional condition is required: the *ModeName* of the current event must differ from that of the previous event to trigger the start of a new leg. This helps to prevent misclassification of vessel transshipments as a single uninterrupted transport-leg.

Although it is technically feasible to analyze transport segments occurring entirely within a terminal, referred to in this study as *intra-terminal legs*,this introduces further complications. Specifically, equipment-related events, which often capture intra-terminal activity, frequently lack a specified *ModeName*. According to the DCSA standard, the omission of this field is permissible. An exploratory analysis of event data reveals that approximately 64.7% of all equipment events contain an empty *ModeName* field.

This incomplete information leads to inconsistencies in how intra-terminal legs are identified. For example, a new leg might be initiated by a discharge event lacking a *ModeName*, while in other instances it may be triggered by a subsequent event. Such ambiguity complicates any systematic analysis of container dwell times at terminals. As a result, equipment events must be excluded from the dataset to ensure consistent leg segmentation. Consequently, this study focuses solely on transport events and restricts its analysis to *inter-terminal legs*.

By filtering out all equipment events, the event log for container ABCD1234567 looks like the one in Table 3.2, where sequential events with the same color belong to the same leg.

Table 3.2: Filtered Event Log of Container ABCD1234567

Event Date-Time	Status	Event	Location	UN/LO Code	Mode	ModeName	Publisher
2025-04-16T15:28:00	ACT	ARRI	MUNDRA	INMUN	TRUCK		Hapag
2025-04-21-T21:18:00	ACT	DEPA	MUNDRA	INMUN	VESSEL	Charleston Express	Hapag
2025-05-28T15:00:00	PLN	ARRI	DDE (Delta)	NLRTM	VESSEL	Charleston Express	ECT
2025-05-30T14:30:00	PLN	DEPA	DDE (Delta)	NLRTM	VESSEL	Charleston Express	ECT
2025-05-30T22:00:00	PLN	DEPA	ROTTERDAM	NLRTM			Hapag
2025-06-02T22:00:00	PLN	ARRI	VENLO	NLVEN			Hapag

3.3. Outlier Detection

An initial attempt at detecting low-quality data was made using outlier detection. There are several ways to measure outliers in the transport legs. In this research a data point is determined as outlier when it is more than 1.5 times the Interquartile Range (IQR) below the first quartile (Q1) or above the third quartile (Q3) of the reference dataset.

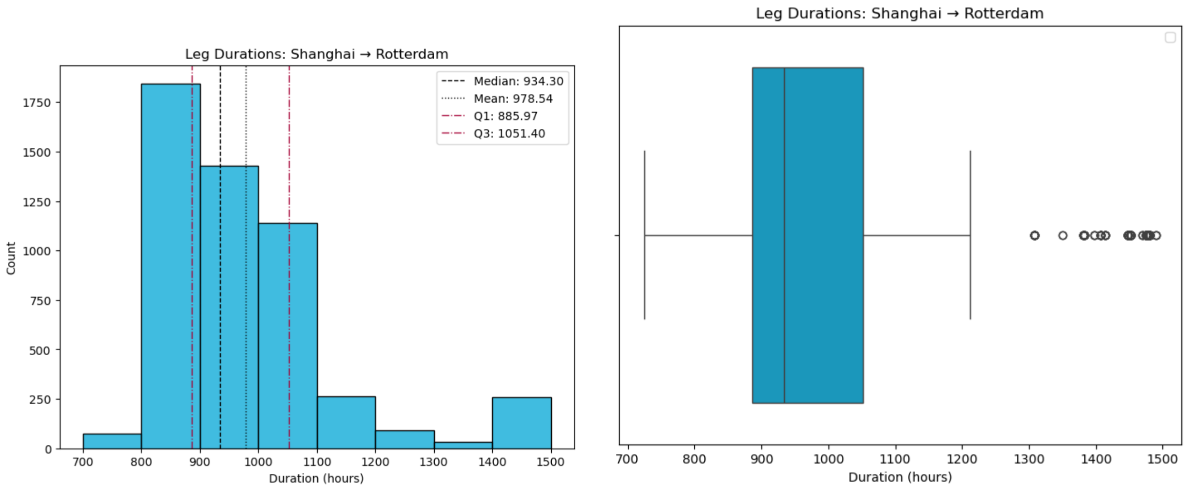
Another common method for detecting outliers is the 3-sigma rule, where data points falling outside three standard deviations ($\pm 3\sigma$) from the mean are considered outliers. However, this method explicitly assumes that data are normally distributed around the arithmetic mean, an assumption often not valid for container transport durations. Container durations typically deviate from a normal distribution, as illustrated by the boxplot in Figure 3.1b. The asymmetry between quartiles (the interval between Q1 and the median significantly exceeds that between the median and Q3) indicates a left-skewed distribution. This observation is further supported numerically: the arithmetic mean is 978.54 hours, while the median is notably higher at 934.30 hours. Such a discrepancy confirms the non-normal distribution of container transit times, as visualized in Figure 3.1a.

Due to this inherent skewness, the 3-sigma rule is less robust, particularly in scenarios involving systemic disruptions or shifts in logistics operations that substantially affect travel times. In contrast, the IQR-based approach provides more reliable and distribution-independent outlier detection, making it better suited for analyzing data with irregular distributions typical in container transportation contexts.

Now that the interquartile range (IQR) method has been identified as a better fit for outlier detection than the traditional 3-sigma rule, it is important to clearly define the approach used to implement it.

For a given origin–destination (OD) pair, the historical records of all containers that have travelled between the two locations are compiled to form a dataset. For each container in this dataset, the duration of the journey, from departure at the origin to arrival at the destination, is calculated. This duration can be either an actual value, if the arrival event is recorded with status `ACT`, or an expected value, if the arrival status is `EST` or `PLN`.

Once the durations for each container are computed, the first quartile (Q_1), third quartile (Q_3), and interquartile range (IQR) can be derived from the training subset of the dataset. These reference values are then used to define a valid range for journey durations, serving as the basis for inlier and outlier classification.



(a) Distribution of durations for Shanghai to Rotterdam

(b) Boxplot of durations from Shanghai to Rotterdam

Figure 3.1: Transport-legs of all direct containermoves from Shanghai to Rotterdam

Let d_j denote the observed duration of container journey j, and define the valid range using the first quartile Q_1, third quartile Q_3, and interquartile range $IQR = Q_3 - Q_1$. Then, the classification function x_j for each journey j is given by:

$$x_j = \begin{cases} 1 & \text{if } Q_1 - 1.5 \times IQR \leq d_j \leq Q_3 + 1.5 \times IQR \\ 0 & \text{otherwise} \end{cases}$$

Here, $x_j = 1$ indicates that the journey is classified as an inlier (regular duration), and $x_j = 0$ indicates that it is classified as an outlier (non-regular duration).

There are several important design choices in how this dataset should be constructed and sampled:

1. **Sampling one container per vessel:** To avoid over-representing vessels carrying a large number of containers, only one container per vessel should be included in the dataset. Otherwise, vessels that transport many containers would disproportionately influence the calculated duration distribution.
2. **Sampling frequency per vessel:** Including the duration of a vessel on every day a container on it is being tracked may introduce bias since the duration for that vessel will likely not change often. Instead, it is necessary to determine how many samples to use per unique vessel. One option is to select only a limited number of representative data points to reduce overrepresentation.
3. **Inclusion of only completed journeys:** Limiting the dataset to vessels that have already arrived provides ground truth values for duration. However, this choice excludes emerging disruptions or anomalies in real-time data, such as events like a Suez Canal blockade, which can delay many vessels similarly and should ideally be captured in the model.

As an initial implementation, the dataset consists of one container per vessel per day. For example, if containers A, B, C, and D are tracked for five days, and containers A, B, and C are on vessel V, while container D is on vessel W, the dataset will include one duration from vessel V and one from vessel W for each day. This results in a total of 10 data points (5 for vessel V and 5 for vessel W).

This method is visualized in Figure 3.2using Buenos Aires as the origin and Rotterdam as the destination. Only containers with an actual recorded arrival in Rotterdam are considered. According to the historical event data, a total of 25 containers completed the Buenos Aires → Rotterdam leg, distributed across 6 distinct vessels. For this dataset, the calculated reference values are: $Q_1 = 715.72$, $Q_3 = 824.30$, and $IQR = 108.58$.

From Figure 3.2, it can be observed that several data points fall below the lower bound defined by

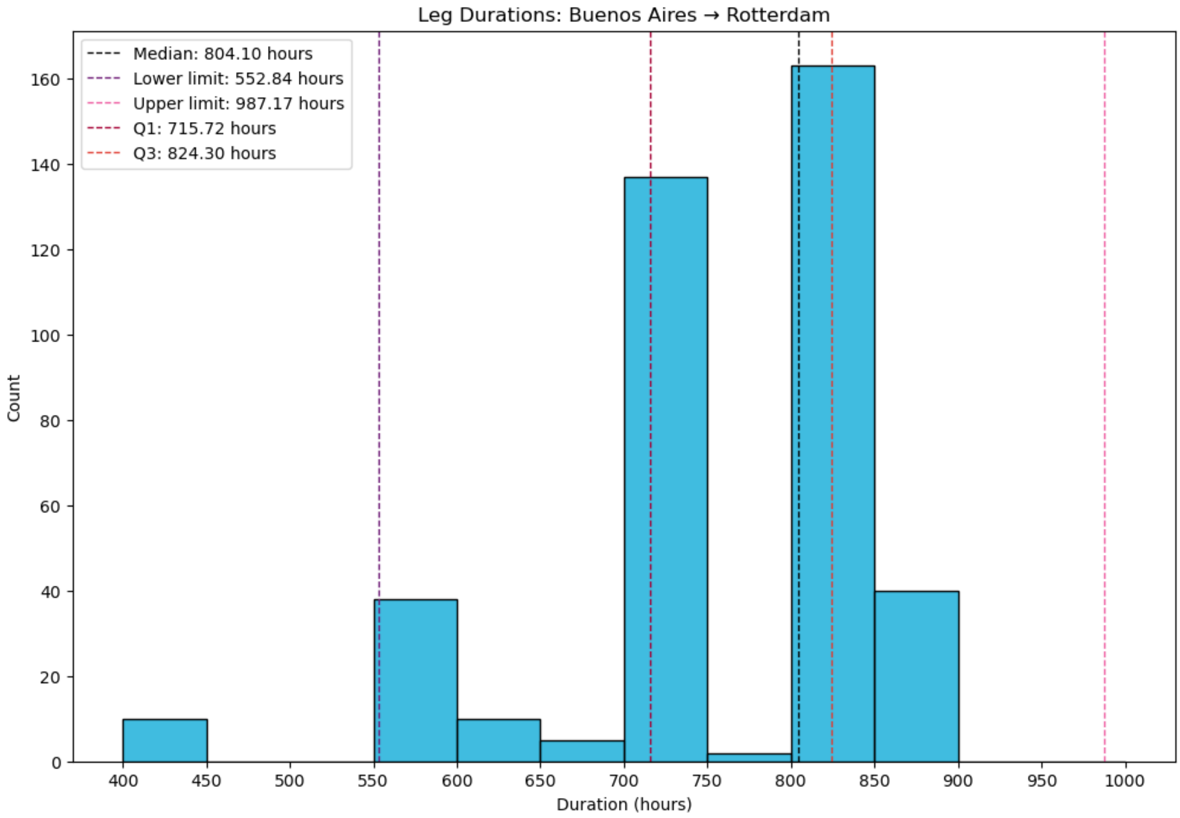


Figure 3.2: Durations for completed transport legs between Buenos Aires and Rotterdam

the IQR method. Upon further investigation, these points correspond to two containers which initially appeared to have a duration of 407.92 hours. However, one week later, updated tracking data extended the duration to 591.3 hours, lifting them above the lower threshold. The evolution of these durations over time is shown in Figure 3.3. It is important to note that we are now referring to containers rather than vessels. Since both containers were aboard the same ship and shared identical event timestamps throughout the tracking period, only one line is visible in the plot for these two outlier containers. The remaining tracked containers did not exhibit durations below the lower bound or above the upper bound during the tracking period. To demonstrate this, the expected duration of two representative containers from different vessels on every day is also plotted.

Broadly speaking, outliers in the data can have two possible causes:

1. Errors in the data (e.g. incorrect vesselname or incorrect Eventtime)
2. Actual operational anomalies (e.g. delays due to congestion or blockages)

Outlier detection (or at least, this method of using outlier detection) can not distinguish between these two and is therefore not suited as a way of detecting data quality issues.

Another issue with outlier detection is that it can not detect inlier container issues. When the IQR is too large, certain inlier containers will not be captured. Figure 3.3 also illustrates this. A DQ question is whether you can predict that a change in duration is going to happen. The outlier container that first took 400 hours (17 days), all of a sudden spiked to around 550 hours (23 days), and can thus be correctly classified as low DQ. However, the inlier container on the left is not captured by outlier detection, but its duration also increased (from 600 hours to 800 hours).

Because of these two reasons another method of detecting DQ issues is necessary.

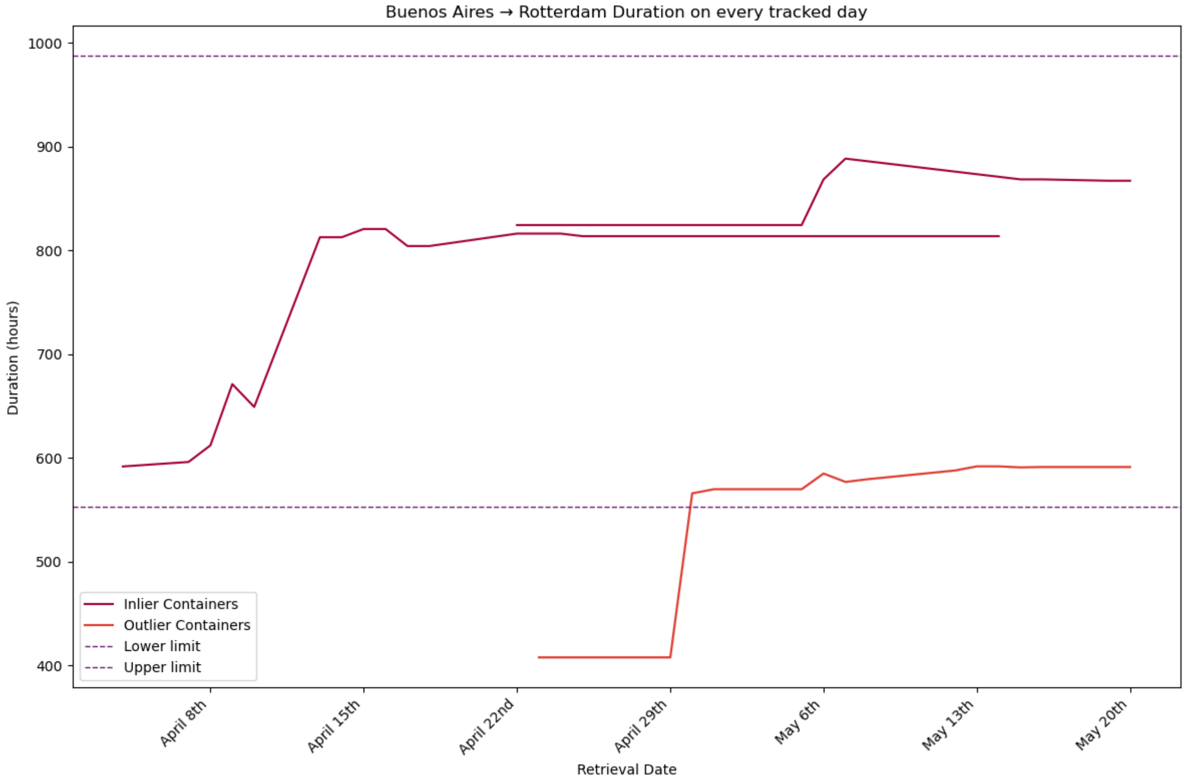


Figure 3.3: Graph of container durations on every tracked day. Note that not all containers between Buenos Aires and Rotterdam are shown.

3.4. Preprocessing for Machine Learning

Another method which can be used to detect low-quality data is with the use of ML. The timestamps of arrivals can be checked by predicting the arrival time, and, assuming that the ML model is able to accurately predict arrivals, when this predicted arrival time and the arrival time in the data correspond, it can safely be assumed that the data is of high quality. However, when the predicted arrival time and the expected arrival time do not correspond, the data points will be flagged as low quality.

Before feeding the eventlog into a ML model, preprocessing needs to take place.

3.4.1. Converting Legs to Tabular Dataformat

First of all, instead of taking the eventlog divided into transport legs, every transportleg is converted to one row. Every row contains the following characteristics of the transport leg:

- Containernumber
- Retrieval date (so on what date this record was recorded)
- Publishing datetime of the departure
- Publishing datetime of the arrival
- The origin
- The destination
- The departure time in the origin
- The arrival time at the destination
- The duration of that leg (departure time substracted from the arrival time)
- The status of the leg (ACT, PLN or EST)
- Which carrier is transporting the container

- The publisher of the arrival event
- The role of the publisher of the arrival event
- How many days the container has been tracked
- And the time between the arrival and the publication datetime of the arrival

An example of such a conversion can be seen in Table 3.3. In there, the journey of container CXDU1789553 is divided into two tranpsort legs. The first from Tianjin Port (CNTXG) to Qingdao (CNQIN), the second leg from Qingdao to Rotterdam (NLRTM). The two legs from the table are retrieved on both May 19th and on June 19th. The leg from Tianjin was already completed when the data was retrieved on May 19th, and as such, the data for that leg is the same on June 19th. However, the leg from Qingdao to Rotterdam are on both retrieval dates not yet completed, and so, as the journey is nearing its completion, the data from June is different than the data from May. From the June data it can be concluded that the vessel left earlier than planned in May, and the arrival is also planned slightly earlier than it was planned in May.

This table alone is not yet enough, since model training requires verified actual times of arrival (ATA). ATAs for each containers's legs were retrieved by scanning the eventlog for records where the status of the arrival event for the container in each leg transitioned from `EST` or `PLN` to `ACT`. Subsequent inspection revealed that some ATAs were retrospectively amended. While $86\%$of such revisions were under 4 hours, a non-negligible amount of containers exhibited corrections that exceeded four hours, with one extreme case of 79 hours (Figure 3.4). To ensure consistency, the *last* recorded ATA for each container was adopted as ground truth. Each row in the table is enriched with its last collected ATA and with the timestamp at which that ATA first became available. This distinction is important, because the last collected ATA may have been available at earlier days too.

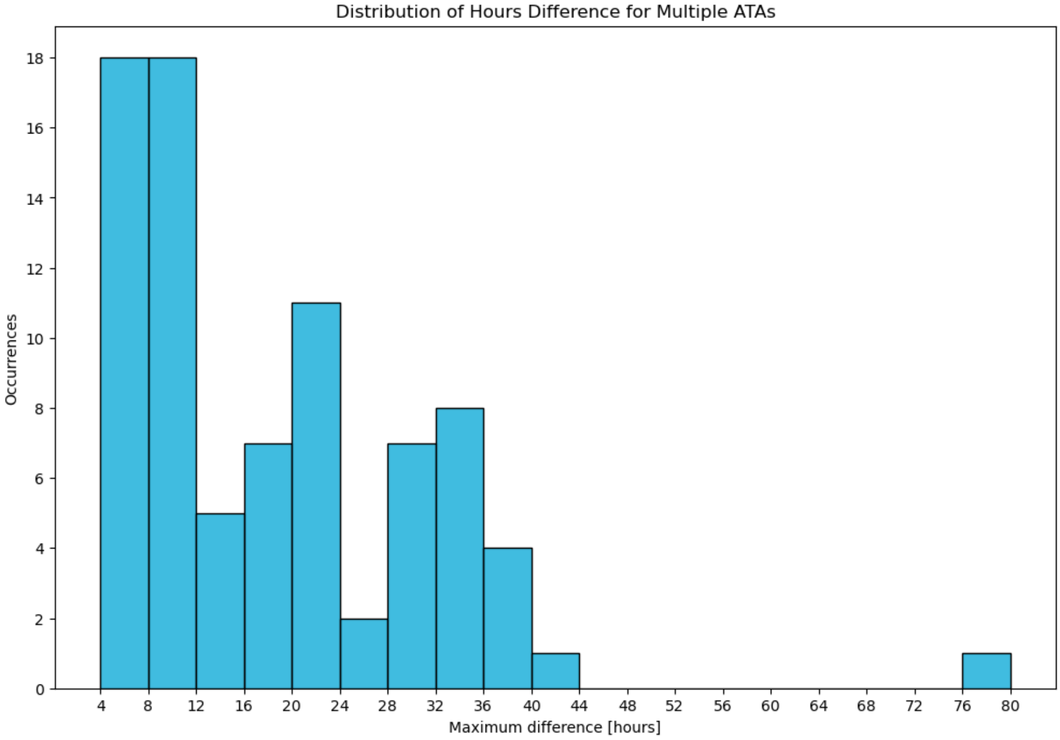


Figure 3.4: Histogram of the maximum time difference between multiple ATAs recorded for the same OD–pair.

3.4.2. Cleaning Invalid UN/LO codes

As part of the initial data exploration, a manual correction procedure was carried out to resolve non-existing UN/LO codes in the dataset. This is important, since the transport leg its origin and destination are based on the UN/LO codesThe top most frequently occurring UN/LO codes were reviewed and cross-checked against the official UN/LO code directory of the UN [20]. In cases where a non-existing UN/LO code was identified, the corresponding location description was used to infer the correct UN/LO code. A mapping table was then created to replace incorrect codes with their correct counterparts. This correction step was essential to avoid misclassifications in origin–destination pairings and to ensure accurate grouping and analysis of container journeys. The full mapping table is provided in Appendix C.

3.4.3. Train-test split

To simulate operations as closely as possible, a time-based train-test split is used, instead of a random train-test split. This means that all data collected upto a certain date is used to train the data, and data collected after that date is used to test the ML models. This strategy mimics the forward-looking nature of real-time deployment and avoids information leakage. In section 4.1.1 the effect of using this split instead of a random split is expanded upon.

After converting all legs in a format usable for the ML regression models, the total amount of legs collected is 89431. Of the 89431 legs, the number of training samples are 72894, while the number of test samples is 16537, this is about equal to an 80/20 train-test split, commonly used in ML problems.

Table 3.3: Example of transport leg data

container	retrieval date	departure published	arrival published	origin	destination	departure time	arrival time	duration [days]	status	carrier	eta publisher	eta publisher role	days tracked	hours before eta
CXDU-1789553	2025-05-19	2025-05-10-T04:36:48	2025-05-13-T20:46:37	CNTXG	CNQIN	2025-05-09-T08:06:00	2025-05-13-T19:12:00	4.46	ACT	CMA CGM	CMA CGM	CA	4	-1.585
CXDU-1789553	2025-05-19	2025-05-16-T14:43:24	2025-05-19-T08:45:52	CNQIN	NLRTM	2025-05-21-T11:00:00	2025-07-10-T17:00:00	50.25	PLN	CMA CGM	RWG	TR	4	1256.24
CXDU-1789553	2025-06-19	2025-05-10-T04:36:48	2025-05-13-T20:46:37	CNTXG	CNQIN	2025-05-09-T08:06:00	2025-05-13-T19:12:00	4.46	ACT	CMA CGM	CMA CGM	CA	35	-1.58
CXDU-1789553	2025-06-19	2025-05-23-T00:00:08	2025-06-19-T00:58:13	CNQIN	NLRTM	2025-05-22-T00:17:00	2025-07-10-T21:00:00	49.86	PLN	CMA CGM	RWG	TR	35	524.03

4 Predicting Arrival Times with Machine Learning

This study adopts supervised machine learning models to classify individual data points as *high-* or *low-quality* on the basis of their predicted arrival times. However, the question remains what high- and low-quality data is. By talking to experts at Poort8, we came to the following definition for high-quality, fit-for-use data.

Definition 1: *Within a DCSA-compliant container-tracking context, data are deemed fit-for-use when the system is capable of alerting stakeholders that an ETA is likely to shift by at least one calendar day.*

Rather than predicting absolute arrival timestamps, which increase as operations progress, the models estimate the duration of each transport leg. Durations seldom exceed two months and therefore this reduces extrapolation error. This extrapolation error becomes significant when using models that are not well-suited for extrapolation, such as tree-based models. This issue is explored more in Section 4.2.2. The predicted duration will then be added to the departure time to construct a prediction for the arrival time. When this arrival time differs by more than one day from the estimated arrival time of the leg, the data record will be flagged as low-quality. This rule enables automated, journey-level plausibility monitoring without reliance on AIS data.

Table 4.1 lists all the features put in the model. Firstly, categorical variables were one-hot encoded to avoid imposing an arbitrary ordinal structure and to enable the models to learn separate effects for each category without introducing spurious distances. As can be read in D the total amount of features after one-hot encoding was 491, of which 459 were OD-pairs. Next, temporal features were converted to seconds to ensure a consistent, numeric representation that preserves relative intervals. Lastly, standardization of numeric and temporal features to zero mean and unit variance was applied to place all variables on a comparable scale, preventing features with larger magnitudes from dominating the model optimization process. These transformations follow established preprocessing practices in machine learning and improve model stability and interpretability.

Occasionally, the publication timestamp of an ETA comes after its `eventDateTime`, yielding a negative `Time before ETA publication`. All observations in which this lead time was negative are removed, as these records should not show up in real-time operations.

Since a time-based split is used, and the dataset only contains records from container journeys in the HeyWim system, there are a few OD-pairs that are only travelled occasionally. This means that some OD-pairs only show up in the trraining dataset or in the test set. If they only occur in the test set, it means that the ML model is not trained on it, and this arrival time can not be predicted accurately. Therefore, an additional filtering step is needed for the test set which is that it can only contain records with an ODpair that is also present in the training set. This filtering step removes around 700 records

Table 4.1: Features used for duration prediction

Feature	Type
Origin–destination (OD) pair	Categorical
Carrier	Categorical
ETA publisher (terminal, carrier, …)	Categorical
Published ETA (expected duration)	Time
Time before ETA publication	Time
Total days tracked	Numeric (days)
Departure timestamp	Time

from the test set, filtering from 16537 samples to 15741 samples. The training set still contains 72894 samples. The impact of this filtering step is further explored in Section 4.1.2.

Now that the preprocessing has been done, the ML models can be trained. The results of testing these models' its ability to predict ATAs is summarized in Table 4.5. All models are implemented in the Python programming language, using the library scikit-learn. Each section provides a table with the hyperparameters used in each model. If there are hyperparameters not mentioned in the table, then the default of scikit-learn was used. The ML models trained in this thesis are:

- **Linear Regression (LR)**: As the least computationally demanding approach, LR enables rapid feature experimentation and establishes a robust baseline for assessing more sophisticated models. The marginal performance improvements offered by more complex techniques may not justify their additional computational cost.
- **Random Forest (RF)**: RF has been extensively employed in ATA prediction using AIS data, and showed promising results (Jiang et al., 2025).
- **XGBoost**: This gradient-boosting method has also yielded promising results in ATA forecasting with AIS inputs, as evidenced by Jiang et al. (2025).
- **Neural Network (NN)**: When AIS data are combined with environmental variables, NNs have achieved one-hour accuracy in ATA prediction (Jahn and Scheidweiler, 2018). Furthermore, Mekkaoui, Benabbou, and Berrado (2022) demonstrated that neural networks outperform other models in this domain.

However, because all prior studies have exclusively utilized AIS data, and sometimes enriching it with environmental data, strong predictive performance in that context does not necessarily translate to comparable effectiveness when applied to event data. However, predicting ATAs with AIS data is the closest analogue to the subject of this thesis.

4.1. Linear Regression

The first predictive model is a Linear Regression model. Owing to its low computational cost, linear regression provides a convenient test-bed for feature experimentation, rapid diagnostics, and transparent coefficient interpretation. Moreover, it establishes a reference point against which any performance gains from more sophisticated algorithms can be meaningfully assessed. LR can be used to test the impact of the methodological choices made in this research.

We proceed in three steps. First we compare a random split with a time-based split, because prediction in this setting is time-ordered and a random split can look better than it should when the same container appears in both sets. Secondly, the time-based split leads to OD-pairs that occur only in the test set; in a dummy-encoded linear model this creates large errors, so we compare two remedies and choose the structural one. Lastly, after fixing the split and the OD-pair coverage, we add two frequency features to test whether they improve accuracy.

4.1.1. Random versus Time-based Train–Test Splits

Firstly, the impact of using a time-based train-test split instead of using a random split is explored. Figure 4.1 displays residuals obtained from a random 80/20 train–test split. Nearly all residuals lie within ±10 days; the model achieves an RMSE of 2.41 days and an R^2 of 0.989, suggesting an almost perfect fit.

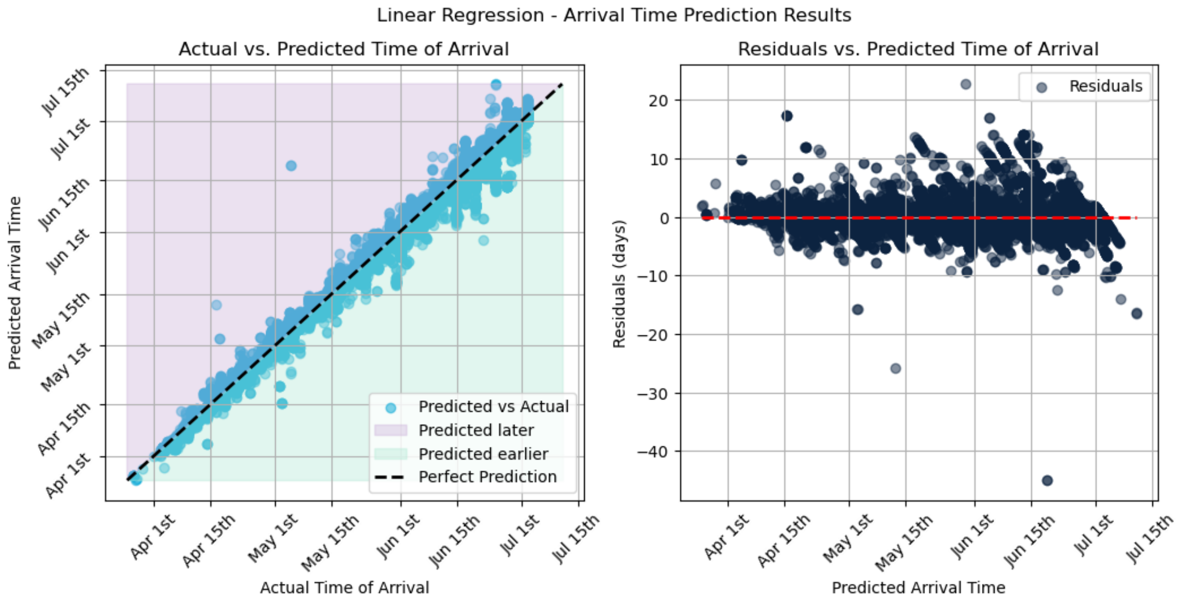


Figure 4.1: Linear Regression performance using a random train–test split.

By contrast, Figure 4.2 shows results obtained with the time-based split. The RMSE increases to 3.25 days and the R^2 drops sharply to 0.381. The apparent visual improvement is therefore an artifact of differing residual scales rather than superior predictive accuracy.

The linear effects in the residuals plots are an artifact of having multiple predictions for a single container. For instance, for a container that arrives on July 1st, during the entire period it is being tracked through HeyWim an arrival time is also predicted. All these predictions are bound to a single arrival time, which means that as predictions get more accurate, the residuals become smaller and smaller for that arrival time.

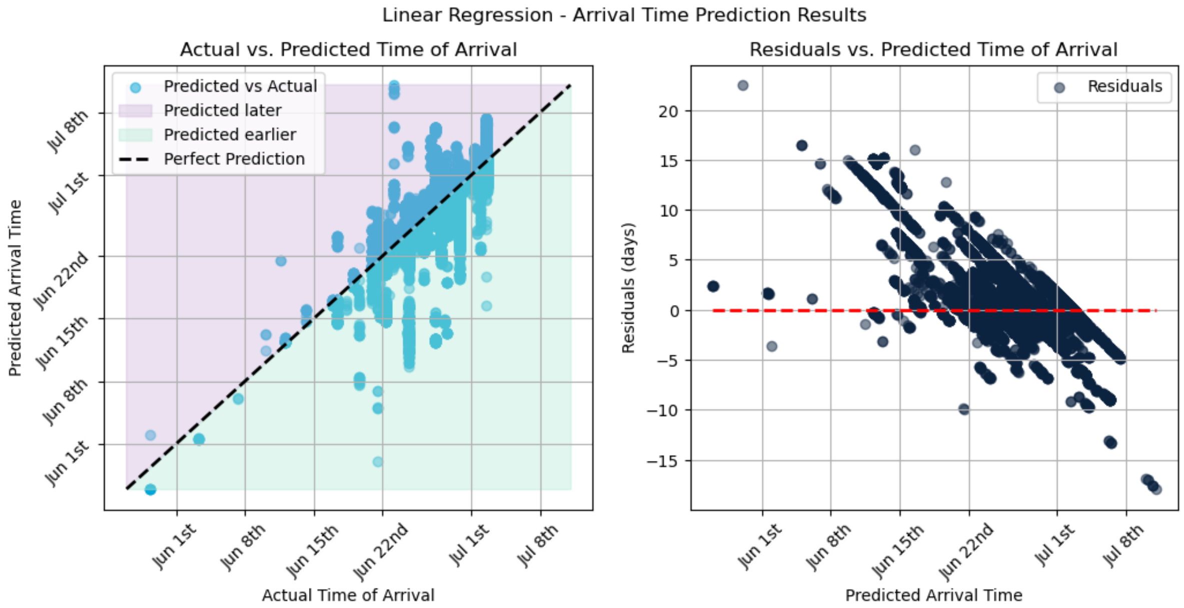


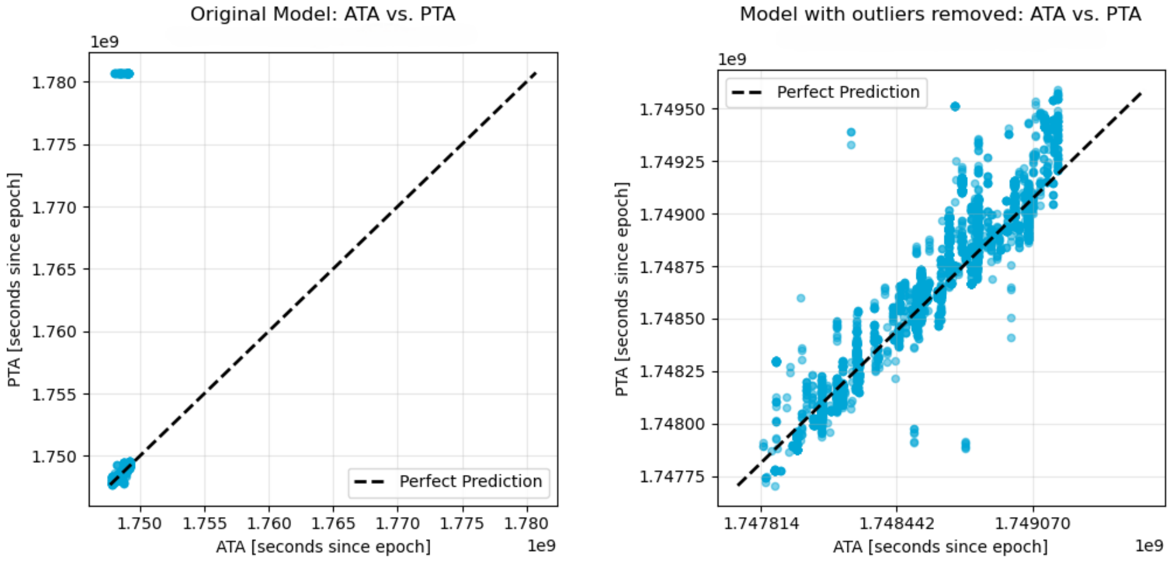
Figure 4.2: Linear-regression performance using a time-based train–test split.

4.1.2. Effect of Unseen Origin–Destination Pairs

A time-based split can yield origin–destination (OD) pairs in the test set that are absent from the training set. Because the associated dummy variables are unseen during training, the model assigns them zero weight. Because OD pair is among the most influential predictors (see Appendix D) this will produce large systematic errors. Without mitigation, the structure appears as in Figure 4.3a, with two clusters. Note that this was done on a previous iteration of the model, so the figures are only illustrative. Two remedial strategies were investigated:

1. **Reactive filtering**: discard predictions whose absolute error exceeds one year. The results (Figure 4.3b) show only one cluster.
2. **Proactive filtering**: restrict the test set to OD-pairs present in the training data, yielding an identical improvement without post-hoc trimming.

Since the outcomes of the two strategies are the same not differ, the most structural approach is the best strategy and thus option 2 is adopted.


Original Model: ATA vs. PTA

Model with outliers removed: ATA vs. PTA

(a) Original: ATA vs. prediction

(b) Filtered: ATA vs. prediction

Figure 4.3: Impact of excluding unseen OD-pairs. Notice the two clusters in Figure 4.3a

4.1.3. Effect of Adding Frequency-Based Features

To assess whether additional context could improve predictive accuracy, two additional features were introduced: *Route Frequency* and *Carrier Route Frequency*.

- *Route Frequency* records the total number of observations for a given OD pair
- *Carrier Route Frequency* tallies the number of observations for that identical OD pair *per carrier*

For example, if an OD pair occurs 40 times, 30 times for Carrier A and 10 times for carrier B, then `Route Frequency` equals 40, while `Carrier Route Frequency` takes the values 30 for records with carrier A and 10 for records with carrier B.

Inclusion of these variables produced only marginal change in the linear-regression baseline: the RMSE decreased from 3.25 to 3.24 days (a reduction of roughly 10 minutes), and the R^2 increased slightly from 0.381 to 0.383. Figure 4.4 illustrates the corresponding residual pattern.

The negligible performance gain suggests that, for a linear model, frequency counts fail to capture additional variance beyond that already explained by existing features.

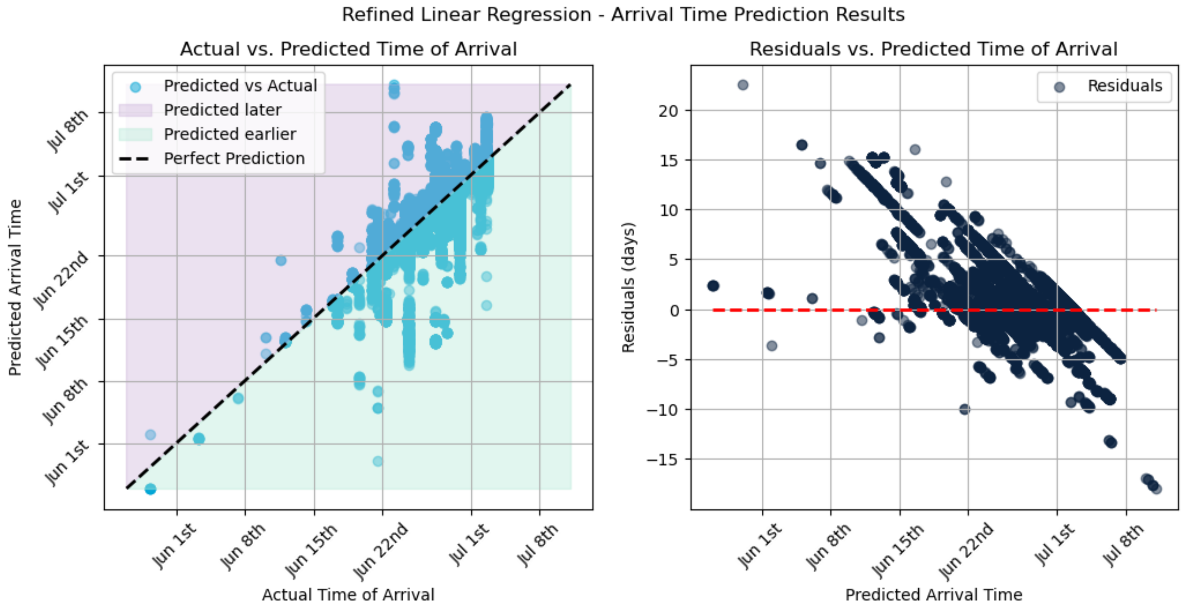


Figure 4.4: Linear regression augmented with frequency-based features.

4.2. Other models

Having established a linear-regression baseline, we next evaluate non-linear algorithms to determine whether predictive accuracy can be improved. The results of all models are plotted just like the example of Figure 4.2. On the left the actual vs predicted times are shown, while on the right a plot of the residuals is shown.

4.2.1. Random Forest

A Random Forest regressor was selected because its ensemble of decision trees can capture complex, non-linear relationships and higher-order feature interactions. For example, an RF can learn that extra delay tends to occur when the `ETA Publisher` equals COSCO *and* the `OD-pair` is 'NLRTM–CNNGB', an effect that is non-additive and difficult to encode manually. After experimenting with the data, the hyperparameters in Table 4.2 are set for the RF model.

Table 4.2: Random Forest hyper-parameters

Parameter	Value
Number of trees	500
Maximum depth	*None* [a]
Min. samples per node	5
Min. samples per leaf	2
Random seed	42

[a] Maximum depth is left at its default value (*None*), meaning each tree Neural-network predictions of leg durationis expanded until all leaves are pure or contain fewer than the minimal samples per node observations, thereby constraining depth implicitly.

Figure 4.5 summarizes the results. The model performs markedly worse than the linear baseline, yielding an RMSE of 6.28 days and an R^2 of -1.308, indicating that this model is not fit-for-use in this regression task.

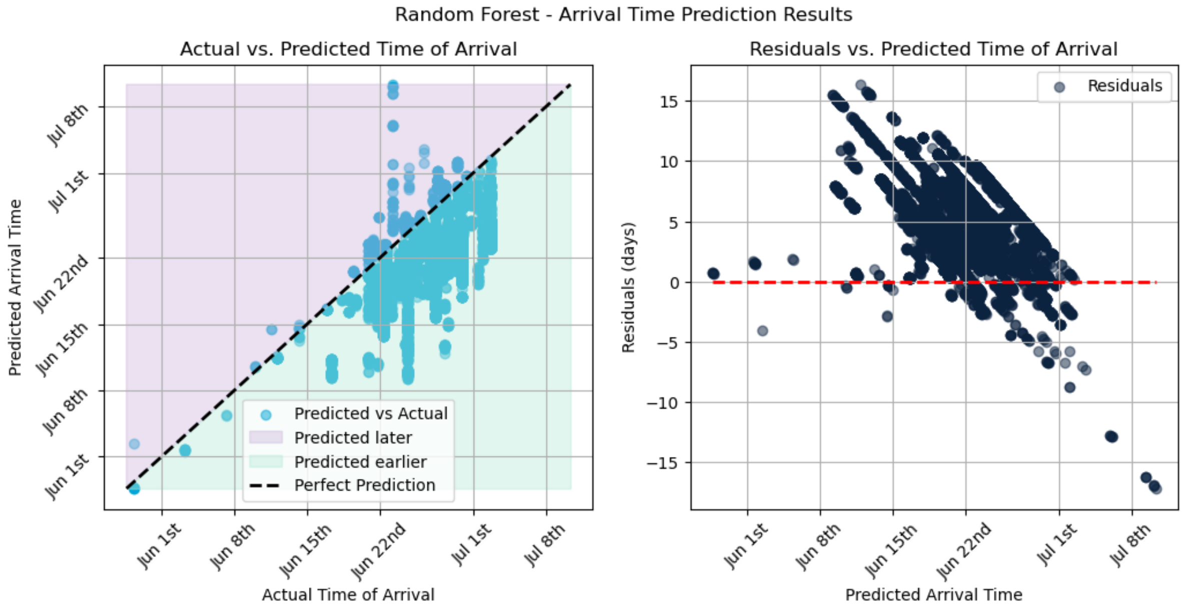


Figure 4.5: Predicting durations with a Random Forest

Because frequency-based variables improved the linear model marginally (Section 4.1.3), the same two features (`Route Frequency` and `Carrier Route Frequency`) were appended to the RF feature set. Accuracy improved but remained inferior to the linear benchmark: RMSE decreased to 5.39 days and R^2 rose to -0.699. Hence, even with additional contextual information, the RF model is not suitable for the present regression task.

4.2.2. Extreme Gradient Boosting

Extreme Gradient Boosting (XGBoost) is a tree-based ensemble method that has performed well in previous ATA-prediction studies [19]. Initial experimentation revealed a fundamental limitation of an earlier approach: predicting arrival times instead of durations. Decision-tree ensembles cannot extrapolate well beyond the numerical range encountered during training. Unlike linear or polynomial regression, trees partition the feature space and return the value stored in one of its leaves. Consequently, an input that lies outside the training range, e.g. an arrival date later than any date seen during training, receives a prediction equal to the historical maximum. In the present data set, the latest training date was June 11th. Figure 4.6 illustrates the resulting horizontal asymptote.

To mitigate this, the target variable was changed from ATA to actual duration, as already explained in the introduction of this chapter. The hyperparameters for the XGBoost model can be found in Table 4.3

Table 4.3: XGBoost hyper-parameters

Parameter	Value
Number of estimators	1200
Learning Rate	0.2
Max. depth	40
Objective	Squared Error
Random seed	42

The duration-based XGBoost model outperformed the RF but remained inferior to LR. Its RMSE is 5.61 days and R^2 is -0.836. Augmenting the feature set with `Route Frequency` and `Carrier Route Frequency` provided some improvement (RMSE = 5.45 days; $R^2 = -0.736$), but not as much of a jump as in the RF model. Figure 4.7 summarizes the final residual pattern. The few points on the bottom suggest that the tree could not fully fit the test data, so training with bigger trees might resolve the fitting

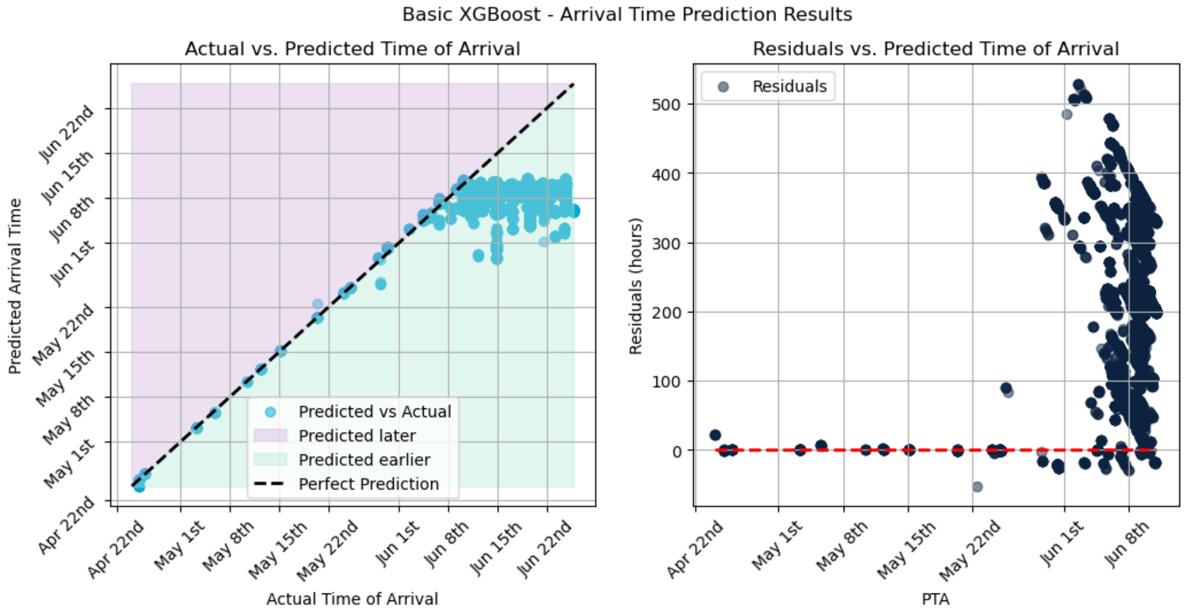


Figure 4.6: Extrapolation failure of XGBoost when trained to predict arrival timestamps; note the plateau at 11 June.

of these points. However, it can also be that a more informative feature already captured the split, such that its sibling never got used.

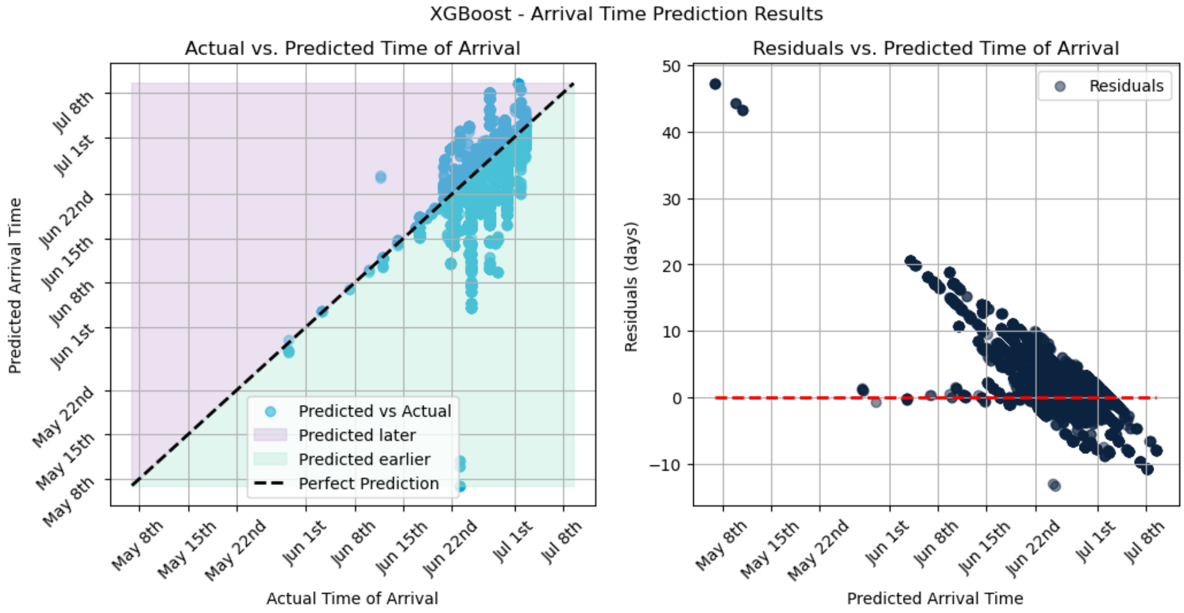


Figure 4.7: Predicting durations with XGBoost

4.2.3. Neural Network

Neural networks are capable of approximating complex, highly non-linear relationships that may exist in the data; however, their predictive power is sensitive to hyper-parameter selection and they require longer training times, making rapid experimentation more costly. After trial and error, the hyperparameters in Table 4.4 seemed like the best performing, so they were used throughout this research.

The baseline NN (Figure 4.8) achieved an RMSE of 4.20 days and an R^2 of -0.030, performing worse than both linear regression and XGBoost. Adding the frequency-based variables further degraded

Table 4.4: Neural-network hyper-parameters

Parameter	Value
Architecture	[40, 30, 40, 20] neurons, ReLU
Optimiser	Adam
Learning Rate	0.001
L2-regularization	0.01
Epochs	2000
Loss function	Squared Error
Random seed	42

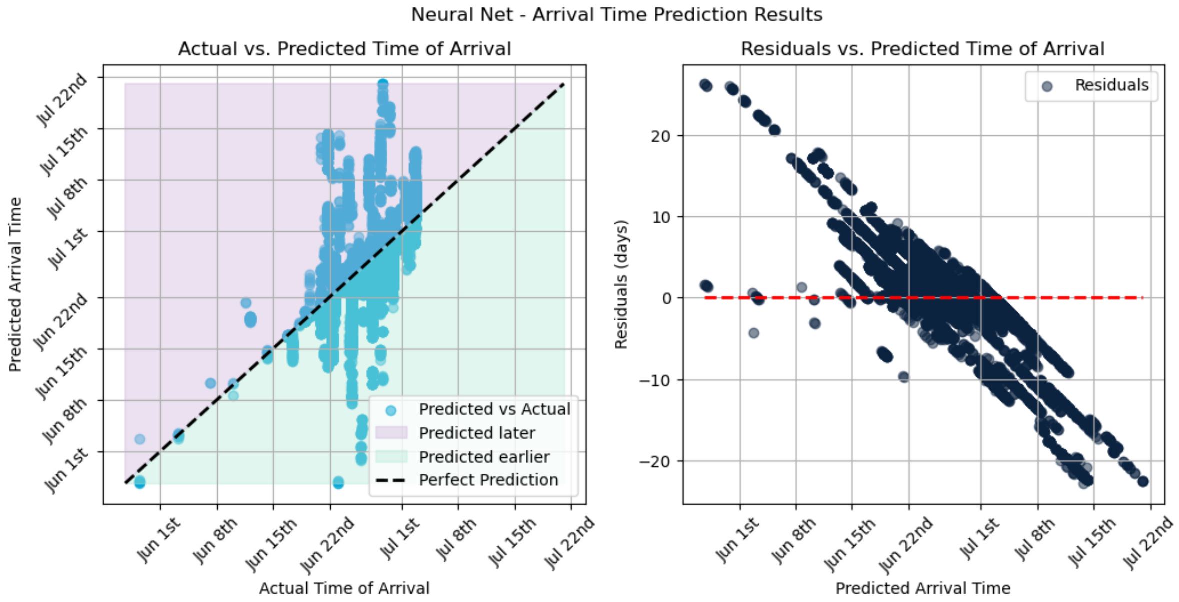


Figure 4.8: Neural-network predictions of leg duration

performance (RMSE = 7.36 days; $R^2 = -2.166$), unlike the LR, RF, and XGBoost models. This NN therefore offers no practical advantage for this regression task.

Table 4.5: Performance of regression models

Model	**Standard**		**With extra features**	
	RMSE (days)	R^2	RMSE (days)	R^2
LR	3.25	0.381	3.24	0.383
RF	6.28	-1.308	5.39	-0.699
XGBoost	5.61	-0.836	5.45	-0.736
NN	4.20	-0.030	7.36	-2.166

4.3. Data quality assessment

While the previous sections focused on accurate regression, the ultimate goal of this research is data-quality assessment. Definition 1 is used in this section to classify each transport leg as either low-quality or high-quality, with the help of the regression models created in the previous sections.

To visualize the process of classification, Figure 4.9 is taken as an example.

The ATA is on 29 June 2025, at 21:00 UTC. The ETA provided in the eventlog always remained earlier than 28 June 2025, 11:00 UTC until the final update, when the ETA was almost spot on. Therefore, all

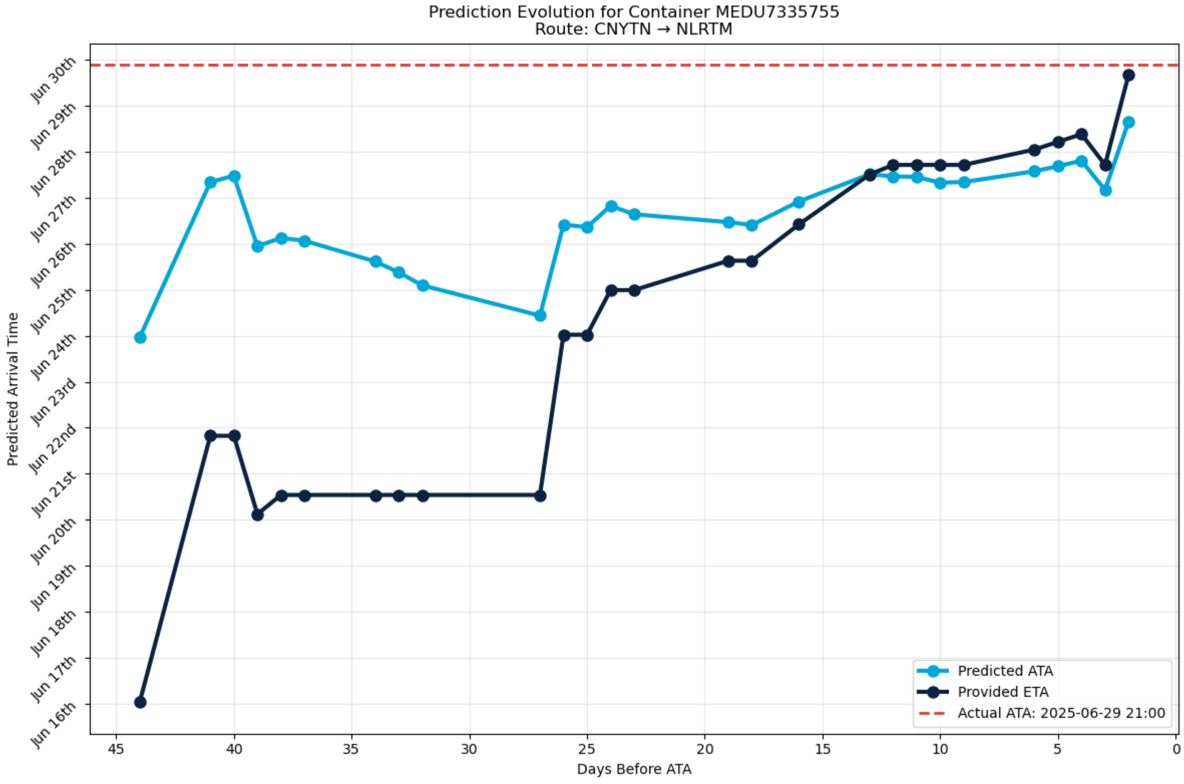


Figure 4.9: Predicted ATAs and ETAs of container MEDU7335755 over time

records, except for the last *should* be flagged as low-quality, according to Definition 1.

Turning to the predicted ATA: From day -30 to day -16 the prediction was closer to the ATA than the ETA provided in the data was. This difference is at least one day, until around day -19. Therefore, all those records were flagged as low-quality data. However, one day later (day -17), the prediction was within a day of the ETA provided in the data. From this point on, all records were not flagged as low-quality, while they should have been flagged as low-quality.

To summarize, 27 records are available for this container. 26 out of these 27 are low-quality, and 1 is high-quality. A total of 14 records were correctly flagged as low-quality. The remaining 13 records were flagged as high-quality, although only 1 of them actually was high-quality.

4.3.1. Aggregate Classification Performance

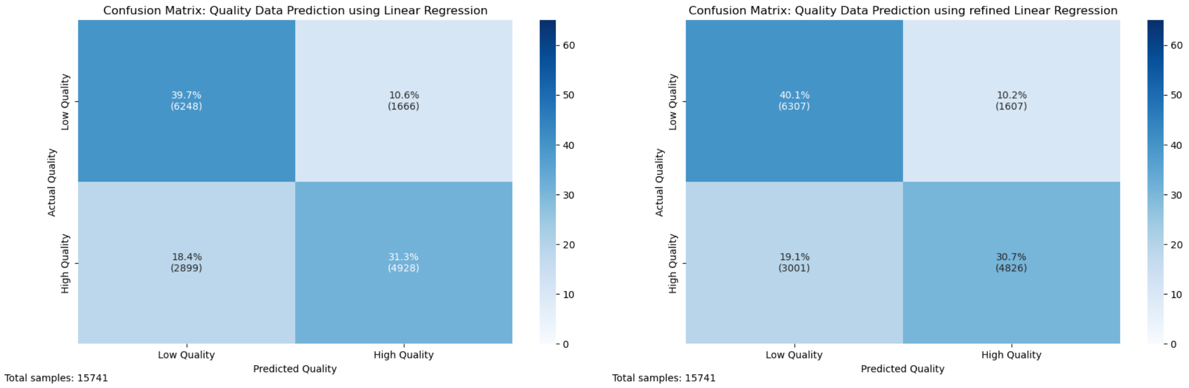
This single-container example illustrates both correct hits and missed detections. The following sections quantify such outcomes across the full data set, reporting True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN) for the LR-model, RF-model, XGBoost-model and NN-model. The interpretation of TPs, FNs, FPs, and TNs are provided in Table 4.6.

Table 4.6: Outcome classes for container-quality flags

Outcome	Flag vs. reality	Interpretation
True Positive	Flagged low-quality, actually low-quality	Hit
False Negative	Flagged high-quality, actually low-quality	Missed
False Positive	Flagged low-quality, actually high-quality	False Alarm
True Negative	Flagged high-quality, actually high-quality	Correct Rejection

Linear Regression

The confusion matrices for the Linear Regression models are shown in Figure 4.10. Linear Regression without extra features is among the best-performing models. Its TP-rate is near that of RF and NN but these models are also raising a lot of false alarms. Linear Regression has the least FP, the highest amount of TNs and only a few FN. Incorporating the extra features, the TP-rate only increased, FN decreased, but the FP increased a bit while the TN decreased. Raising false alarms is not desirable, since it suggests that something is wrong, while nothing is wrong in reality. Missing a few datapoints (FNs) is less of a problem, since doing nothing (which is the current situation) is simply missing all data. Thus, the confusion matrices indicate that adding frequency features degrades LR performance.



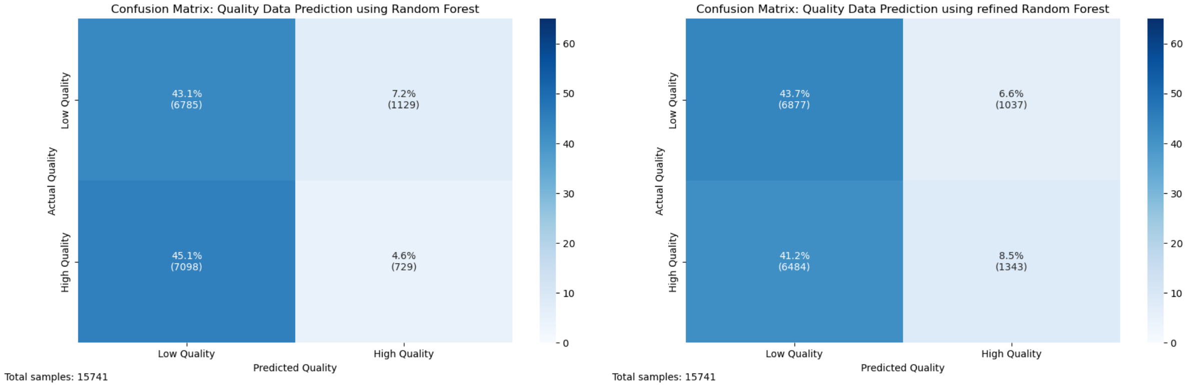
(a) Linear Regression

(b) Linear Regression with extra features

Figure 4.10: Confusion Matrices for Linear Regression

Random Forest

The confusion matrices for the RF models are shown in Figure 4.11 The Random Forest model clearly has a tendency to predict ATAs that are at least a day apart from the ETA. This makes it that the model captures a lot of accurately predicted low-quality datapoints, but it also raises even more false alarms. Incorporating extra features did lower the amount of low-quality predictions, but since there is still such a large amount of FPs, it seems like Linear Regression is the better choice.



(a) Random Forest

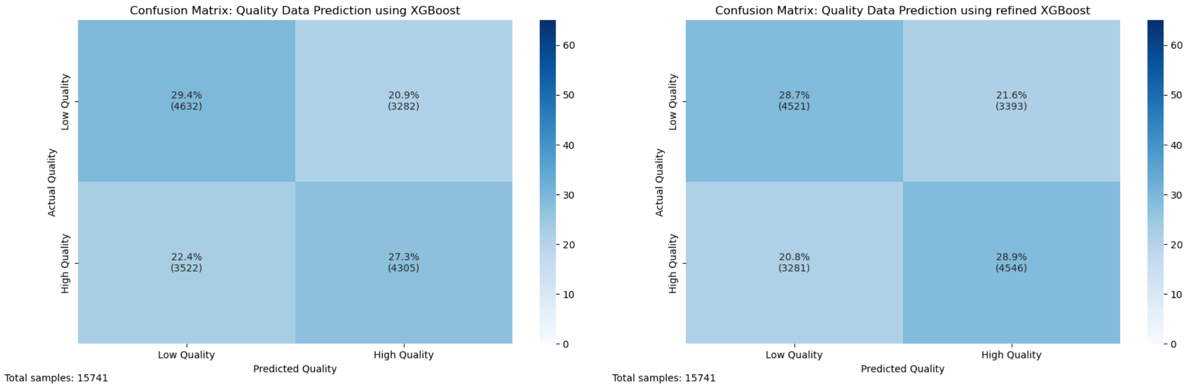
(b) Random Forest with extra features

Figure 4.11: Confusion Matrices for Random Forest

XGBoost

Using the regressions of the XGBoost model and then classifying the data as either low-quality or high-quality provides confusion matrices as shown in Figure 4.12. This already shows more balance in high-quality and low-quality, so false alarms are raised not as often as with the RF models, which is good. Adding extra features lowers the amount of low-quality predictions, which in turn lower the amount of false alarms even more. However, the amount of false alarms is slightly higher than it is with

Linear Regression, and the amount of missed data points is substantially higher. Therefore, XGBoost looks like it is performing not as good as Linear Regression is.



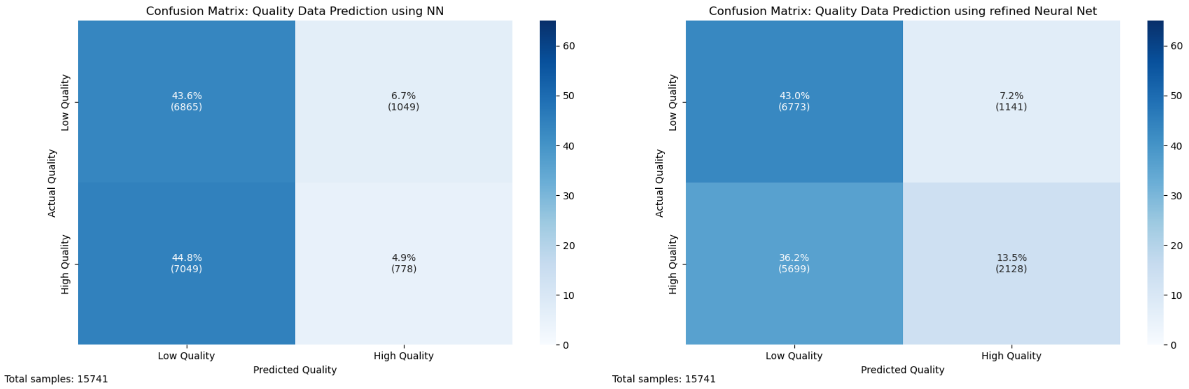
(a) XGBoost

(b) XGBoost with extra features

Figure 4.12: Confusion Matrices for XGBoost

Neural Network

The confusion matrices for the NNs (Figure 4.13) show a similar pattern as the confusion matrices for RF; both have a tendency to often predict low-quality data. The amount of false alarms here is very high, and adding extra features does help a bit, but not enough to match the performance of XGBoost or LR.



(a) NN

(b) NN with extra features

Figure 4.13: Confusion Matrices for NN

Overall Model Ranking

For implementation of this process in production, it is of utmost importance that whenever a leg is flagged as low-quality, this is also actually the case. So in an ideal world, no false alarms are raised (i.e. the number of FPs is 0). However, this would reward models that only predict high-quality data. Therefore, a metric that captures good model performance should also include the number of TPs, so that it will actually predict low-quality data. A metric that takes into account both TPs and FPs is the Positive Prediction Value (PPV), also known as precision. The definition for PPV is $PPV = \frac{TP}{TP+FP}$. However, if positives are rare, the PPV will be high, and so this metric would still not be able to capture the performance well of models that predominantly predict high-quality data. Such a model would contain a lot of misses (FNs) which is also undesirable. However, since avoiding false alarms is of such high importance, the model should rather make a few FNs, than make a single FP.

Therefore, to make a decision on what model performs best, a KPI is needed that rewards:

- High purity of positive predictions (i.e. few FPs)

- High reliability of negative predictions (i.e. few FNs, but these matter less than FPs)

A KPI that checks all these boxes is the generalized F-score. The formula for the F-score is as follows:

$$F_\beta = (1+\beta^2) \cdot \frac{\text{precision} \cdot \text{recall}}{(\beta^2 \cdot \text{precision}) + \text{recall}} \tag{4.1}$$

When setting β to 1, F_1 is the harmonic mean of precision (PPV) and recall. Precision measures how many of the items labelled "positive" are actually positive, where recall measures how many of the true positives are successfully caught. Recall is defined as $\frac{TP}{TP+FN}$ The F_1-score combines recall and precision into one number by taking their harmonic mean, so you only get a high F-score if both precision and recall are high. By setting a different value for β, the importance of precision and recall relative to each other can be set.

Speaking to experts at Poort8, it was decided that for every 50 FNs (misses) a single FP (false alarm) could be accepted. So, the 'cost' of a FP is 50, while the 'cost' of a FN is 1. β can then be calculated according to the following formula: $\beta = \sqrt{\frac{C_{FP}}{C_{FN}}}$, where C_{FP} and C_{FN} are the cost of a FP and a FN, respectively. Substituting in the cost values of a FN and a FP, provides a β of $\sqrt{\frac{1}{50}} \approx 0.141$.

The $F_{0.141}$ score for all models is provided in Table 4.7, alongside the number of TPs, FNs, FPs and TNs. The $F_{0.141}$ score for standard Linear Regression is the best, as looking at the confusion matrices already suggested, and now is confirmed by the $F_{0.141}$ score.

Table 4.7: Classification metrics for each model ($\beta = 0.141$)

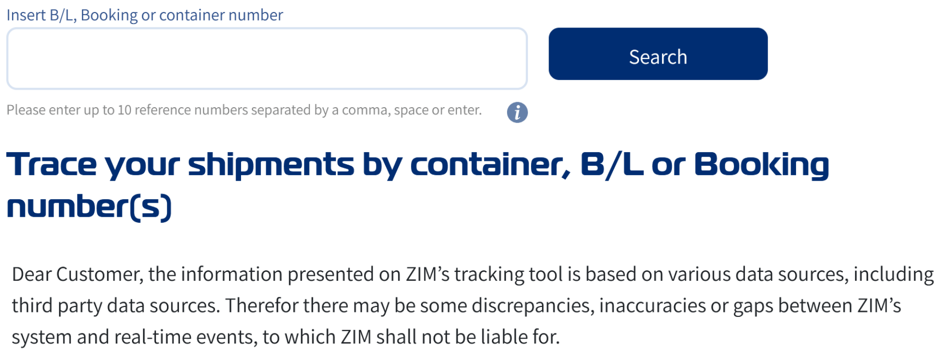
	Model			
Metric	Linear Regression	Random Forest	XGBoost	Neural Network
		Standard		
TP	39.7%	43.1%	29.4%	43.6%
FN	10.6%	7.2%	20.9%	6.7%
FP	18.4%	45.1%	22.4%	44.8%
TN	31.3%	4.6%	27.3%	4.9%
PPV	68.3%	48.9%	56.8%	49.3%
$\mathbf{F}_\beta$	**68.5%**	**49.3%**	**56.8%**	**50.0%**
		With extra features		
TP	40.1%	43.7%	28.7%	43.0%
FN	10.2%	6.6%	21.6%	7.2%
FP	19.1%	41.2%	20.8%	36.2%
TN	30.7%	8.5%	28.9%	13.5%
PPV	67.8%	51.5%	57.9%	54.3%
$\mathbf{F}_\beta$	**68.0%**	**51.9%**	**58.0%**	**54.7%**

4.3.2. Carrier-only Evaluation: Detecting Early-stage ETA Errors

Terminal-sourced ETAs tend to be more reliable than carrier-generated ETAs, because terminals can schedule berth windows with high accuracy. By contrast, carriers aggregate information from heterogeneous and sometimes conflicting sources; several carriers even display explicit disclaimers regarding data accuracy, as shown in Figure 4.14.

Figure 4.9 illustrates that the data in the beginning of a container's journey is highly susceptible to change, while in the later stages of a container its journey the predictions do not really matter anymore, as the difference between ETA and prediction is negligible.

This is further supported by aggregating over all containers and looking at the mean error between the ETA and the ATA, as a function of the lead time to arrivals, plotted in Figure 4.15. Substantial variation is



Please enter up to 10 reference numbers separated by a comma, space or enter.

Trace your shipments by container, B/L or Booking number(s)

Dear Customer, the information presented on ZIM’s tracking tool is based on various data sources, including third party data sources. Therefor there may be some discrepancies, inaccuracies or gaps between ZIM’s system and real-time events, to which ZIM shall not be liable for.

Figure 4.14: Accuracy disclaimer displayed on the ZIM customer portal.

evident in earlier periods, while the average discrepancy narrows to zero after approximately 20 days.

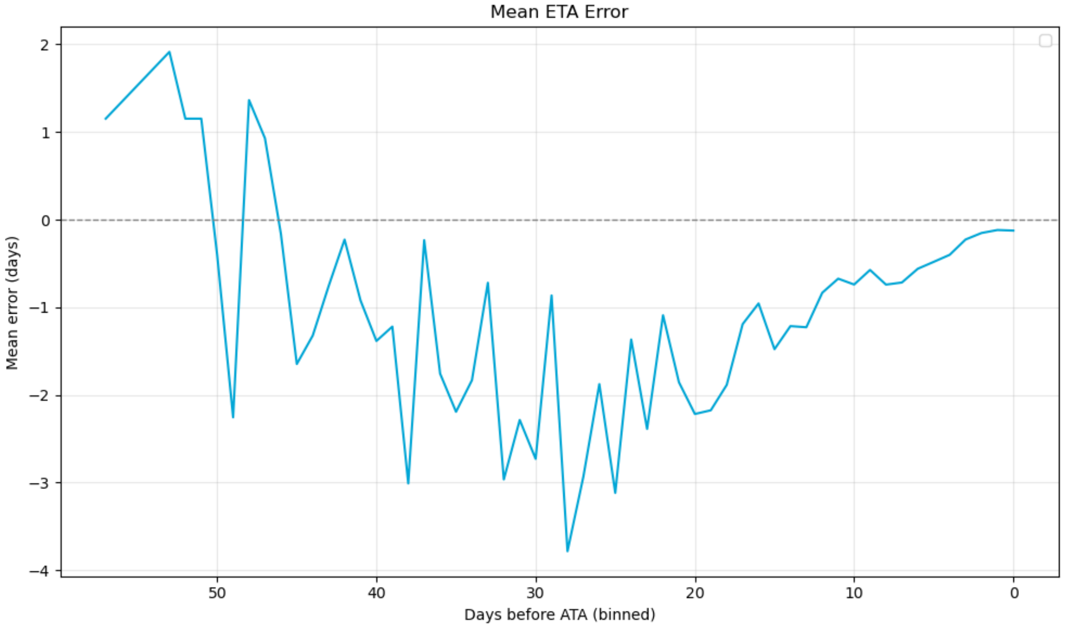


Figure 4.15: Mean ETA–ATA error as a function of days before arrival.

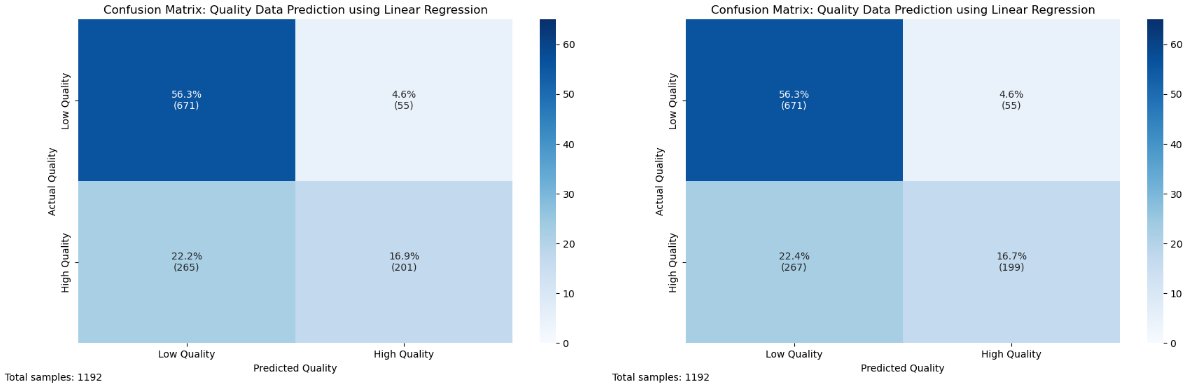
Consequently, in all subsequent analyses only expected arrival events that are published at least 7 days before the ATA and originate from carrier sources are subjected to the high/low-quality classification scheme. Only 1192 observations meet these criteria, so the class balance differs markedly from the full-set analysis.

Linear Regression

Figure 4.16 displays the confusion matrices after restricting the test set to carrier-published ETAs that precede the actual arrival by at least a week. LR now captures even more of the low-quality records, it misses barely anything. Adding the frequency features offers again no significant benefit.

Random Forest

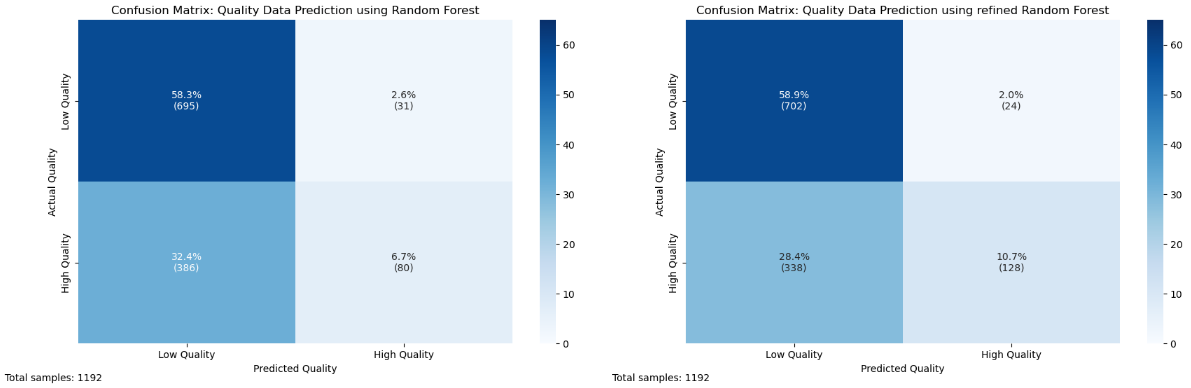
With no extra features the forest flags almost 60% of genuinely bad legs (up from 43% in the full data) while its false-alarm rate drops from roughly 45% to 32%. Adding the two frequency variables increases the performance, although the RF variants still have a bias toward predicting low-quality, so the absolute number of false alarms remains much higher than for the linear baseline.



(a) Linear Regression for carriers only

(b) Linear Regression with extra features for carriers only

Figure 4.16: Confusion Matrices for Linear Regression filtering on carriers and one week before estimated arrival



(a) Random Forest for carriers only

(b) Random Forest with extra features for carriers only

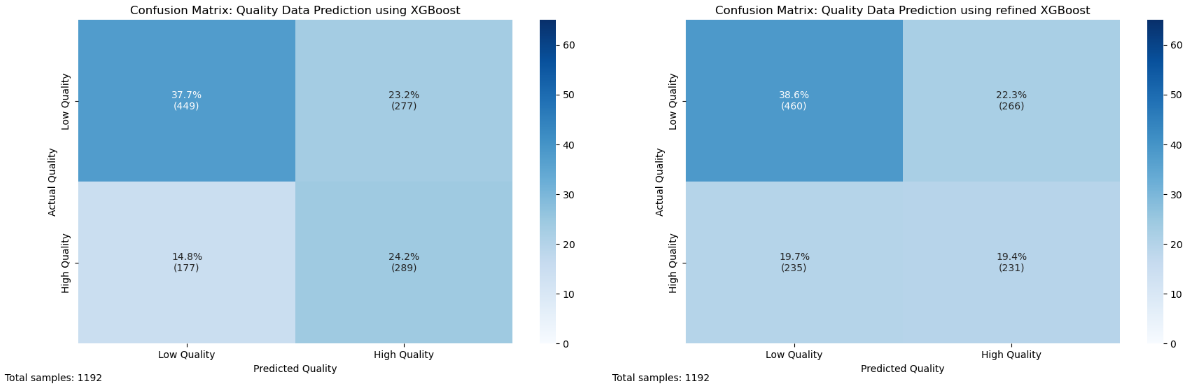
Figure 4.17: Confusion Matrices for Random Forest filtering on carriers and one week before estimated arrival

XGBoost

Over the carrier-only sample XGBoost behaves much as it did on the full test set: its calls are noticeably more balanced than the RF. In the standard model (Figure 4.18a) the number of false alarms is the lowest of all models, however, the number of missed classifications is the highest of all models. In regards to the LR models, the number of hits is also much lower, so this model performs worse than LR. Adding the frequency features (Figure 4.18b) decreases the performance of the model. The number of false alarms increase, and it almost becomes a 50/50 chance whether a record is correctly classified as high- or low-quality.

Neural Network

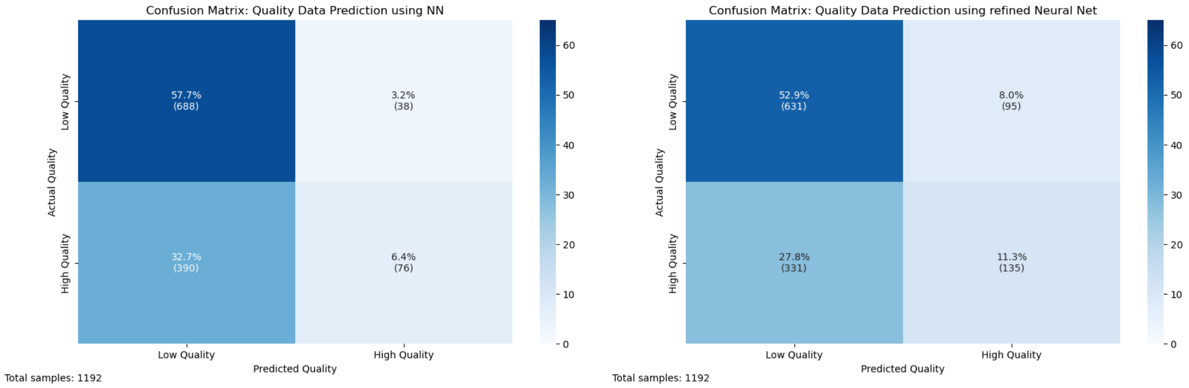
Just like in the classification of the full set, NNs have a bias to predicting low quality, as can be seen in Figure 4.19. 32.7% false alarms are raised as a consequence of this. Adding extra features lowers the amount of low-quality predictions, reducing the amount of false alarms, but also decreasing the amount of hits.



(a) XGBoost for carriers only

(b) XGBoost with extra features for carriers only

Figure 4.18: Confusion Matrices for XGBoost filtering on carriers and one week before estimated arrival



(a) Neural Net for carriers only

(b) Neural Net with extra features for carriers only

Figure 4.19: Confusion Matrices for Neural Net filtering on carriers and one week before estimated arrival

Model Ranking for Early-Stage Carrier ETAs

Just like in Section 4.3.1, Table 4.8 summarizes the key metrics.

For carrier data the best performer is again the standard LR model, although the variant with frequency features trails behind by only a tenth of a percentage point. It is interesting to note that the PPV is almost identical as the $F_{0.141}$ score, illustrating that missed detections carry far less weight than false alarms under the chosen cost ratio. Using PPV as the KPI ties standard XGBoost with Linear Regression, however, LR detects many more low-quality legs (higher TP) and misses fewer low-quality legs (lower FN), so the $F_{0.141}$ metric justifiably gives LR the edge.

Table 4.8: Classification metrics for carriers ($\beta = 0.141$)

	Model			
Metric	Linear Regression	Random Forest	XGBoost	Neural Network
		Standard		
TP	56.3%	58.3%	37.7%	57.7%
FN	4.6%	2.6%	23.2%	3.2%
FP	22.2%	32.4%	14.8%	32.7%
TN	16.9%	6.7%	24.2%	6.4%
PPV	71.7%	64.3%	71.7%	63.8%
$\mathbf{F}_\beta$	**72.0%**	**64.7%**	**71.6%**	**64.2%**
		With extra features		
TP	56.3%	58.9%	38.6%	52.9%
FN	4.6%	2.0%	22.3%	8.0%
FP	22.4%	28.4%	19.7%	27.8%
TN	16.7%	10.7%	19.4%	11.3%
PPV	71.5%	67.5%	66.2%	65.6%
$\mathbf{F}_\beta$	**71.9%**	**67.9%**	**66.2%**	**65.9%**

5 Discussion

This study represents an initial investigation into the prediction of Actual Time of Arrival (ATA) using event data formatted according to the DCSA standard. While voyage-based prediction employing AIS data has been extensively explored and is relatively mature, the present approach, relying solely on eventlogs, constitutes a novel contribution. Future research could enhance predictive performance by incorporating additional features that capture temporal dynamics, such as Suez Canal closures or port congestion. Moreover, a hybrid methodology that synthesises event data with AIS-derived features (e.g., geospatial trajectories) would merit rigorous evaluation.

Also, the weather plays a big role in the arrival of ships, so taking into account weather conditions is key in predicting accurate ATAs. This would also be something to look into in further research.

In this thesis, data quality assessments were conducted through binary judgements regarding whether observations met a minimum standard. An alternative strategy involves training a supervised classifier, such as a support vector machine, to distinguish between high- and low-quality records. Investigating this approach may yield more nuanced insights into the characteristics of inadequate data and improve overall model robustness

Certain OD-pairs appear infrequently in the training data. This scarcity likely limits the model's predictive capability for underrepresented routes. Therefore, restricting analysis to well-trained, high-frequency OD pairs or developing techniques to augment sparse routes (e.g., transfer learning) represents an important avenue for advancement.

Not only scarce data can be a cause of lackluster performance, it is plausible that certain corridors or particular operators exhibit more predictable patterns. Disaggregating results by route or carrier in follow-up research could uncover such heterogeneity and inform targeted model improvements.

Comparable temporal-event-prediction problems may exist in other industries. Although a comprehensive review is beyond this thesis's scope, analogous work in healthcare, predicting timestamps of clinical events, could offer transferable methodologies or evaluation frameworks.

The training dataset spans only a three-month period, during which the Suez Canal remained blocked until near the study's conclusion (Jumelet, 2025). This temporal dependency compromises the statistical independence of training and testing samples. To mitigate this bias, future work should extend data collection across longer intervals or introduce features that explicitly models such disruptions. Another way could be to introduce meta-features that capture disruptions; for example average durations in the last week.

Once such drift-sensitive meta-features are in place, a natural extension is meta-learning. This trains the model itself to adapt rapidly when those features signal a distributional shift. Gradient-based frameworks such as Model-Agnostic Meta-Learning pre-train a shared set of weights on many "tasks" (e.g., weekly corridor slices) so that only one or two gradient steps on fresh data are sufficient to regain accuracy.

Duration predictions were evaluated by comparing forecasted voyage durations to actual ATAs; however, this approach conflates departure time updates with arrival forecasts. In practice, if a vessel's departure is delayed (e.g., by one week), a correct duration estimate may nonetheless yield arrival prediction of a week earlier. Future evaluations should decouple duration accuracy from departure time variability by comparing predicted durations directly to observed durations, irrespective of departure time revisions.

This methodology does not address all dimensions of data quality. For instance, it presumes location reports are accurate. Engaging the client to define and prioritise quality dimensions would clarify the requirements for truly "fit-for-use" datasets. In fact, the concept of *fit-for-use* recurs throughout this thesis. Ultimately, however, the determination of fitness lies with the client; this thesis adopts a one-day discrepancy threshold as the criterion. Incorporating direct client feedback into the fitness evaluation process would provide a more defensible, application-driven standard.

The present analysis focuses exclusively on inter-terminal eventlogs. Investigating intra-terminal events, such as gate processing or yard handling, could reveal additional predictors of ATA and merits further study.

Neural networks typically excel with large volumes of data. Although the current dataset is substantial, its sufficiency for deep learning remains uncertain. As the time horizon of data collection expands, the potential of neural architectures should be evaluated alongside traditional models.

Beyond the statistical and machine learning techniques applied here, process-mining methodologies could uncover anomalous event sequences within container logs (e.g., detecting occurrences where a *Gate In* event immediately follows a *Load* event). Applying conformance checking might enhance anomaly detection and data validation efforts.

Large Language Models (LLMs) may possess the capacity to identify intricate patterns indicative of logical routing sequences. This thesis does not explore such approaches, yet examining their applicability represents a promising direction for future work.

Finally, this research trained each predictive model only once with a set random seed, to ensure reproducibility. In a production context, however, repeated training runs with varying random seeds would yield distributional performance measures and guard against overestimating the efficacy of a particular model instance (e.g., linear regression marginally outperforming XGBoost). Aggregating results over multiple runs would deliver more reliable model comparisons.

6 Conclusion

This research addressed the systematic assessment of data quality in event-based container tracking, particularly within container trackers that are DCSA-compliant. Given the vital role of accurate and timely container tracking in global logistics, ensuring robust DQ is essential. The primary contribution of this study lies in developing a novel approach to assessing data quality based on predicting deviations in ETAs.

The thesis first identified the limitations of traditional dimensional metrics and proposed a ”fit-for-use” criterion defined through consultation with industry experts: data is considered high-quality if ETA deviations do not exceed one calendar day.

Next it showed the process of creating an eventlog from single events, and how this eventlog can be processed to create a single data record for every transport leg.

Machine learning models (Linear Regression, Random Forest, XGBoost, and Neural Networks) were consequently applied to predict ATAs for each transport leg. When this predicted ATA was at least one day apart from the ETA provided for the leg, this record was flagged as low-quality. This enabled a systematic identification of any transport leg.

Among the evaluated models, Linear Regression emerged as the most effective, demonstrating superior precision and reliability in classifying data quality. Notably, more complex models like Random Forest and Neural Networks underperformed relative to the simpler linear model, emphasizing the importance of meticulous preprocessing and domain-specific feature engineering over algorithmic complexity. XGBoost came quite close to the LR, and it had the least amount of false positives, but this came at the cost of having more false negatives, and also less true positives.

When focussing purely on ETAs provided early by carriers, the perfomance of the models increased a bit, which confirms the hypothesis that ETAs provided early in the journey and ETAs provided by carriers are less reliable than ETAs provided by terminals.

Nevertheless, the study faced constraints such as a limited temporal dataset, reliance solely on eventlog data without real-time AIS or environmental variables, and the challenge of sparse data for specific origin-destination pairs. Future work should address these limitations by integrating broader data sources, exploring hybrid methods combining event data with AIS trajectories, and extending the time frame to enhance model generalizability.

In conclusion, this thesis demonstrates the feasibility of systematically improving data quality in container tracking through predictive analytics, offering a concrete methodological advancement for industry practitioners and paving the way for further research in this promising area.

References

[1] Elena Camossi, Tatyana Dimitrova, and Aris Tsois. “Detecting Anomalous Maritime Container Itineraries for Anti-fraud and Supply Chain Security”. In: *2012 European Intelligence and Security Informatics Conference*. IEEE, Aug. 2012, pp. 76–83. DOI: 10.1109/eisic.2012.39.

[2] Lionel A. Galway and Christopher Hanks. *Data Quality Problems in Army Logistics: Classification, Examples, and Solutions*. Santa Monica, CA: RAND Corporation, 1996.

[3] Peter Burggräf et al. “Data quality-based process enabling: Application to logistics supply processes in low-volume ramp-up context”. In: *2018 International Conference on Information Management and Processing (ICIMP)*. 2018, pp. 36–41. DOI: 10.1109/ICIMP1.2018.8325838.

[4] DCSA. *About Us*. Apr. 2025. URL: https://web.archive.org/web/20250429102419/https://dcsa.org/about-us.

[5] Jiarui Xie, Lijun Sun, and Yaoyao Fiona Zhao. “On the Data Quality and Imbalance in Machine Learning-based Design and Manufacturing—A Systematic Review”. In: *Engineering* 45 (2025), pp. 105–131. ISSN: 20958099. DOI: 10.1016/j.eng.2024.04.024. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85213893105&doi=10.1016%2fj.eng.2024.04.024&partnerID=40&md5=030e8231d09d12703f4a7fd9886f5814.

[6] Hong Chen et al. “A Review of Data Quality Assessment Methods for Public Health Information Systems”. In: *International Journal of Environmental Research and Public Health* 11.5 (May 2014), pp. 5170–5207. ISSN: 1660-4601. DOI: 10.3390/ijerph110505170.

[7] Benjamin T. Hazen et al. “Data quality for data science, predictive analytics, and big data in supply chain management: An introduction to the problem and suggestions for research and applications”. In: *International Journal of Production Economics* 154 (Aug. 2014), pp. 72–80. ISSN: 0925-5273. DOI: 10.1016/j.ijpe.2014.04.018.

[8] Richard Y. Wang and Diane M. Strong. “Beyond accuracy: What data quality means to data consumers: JMIS”. English. In: *Journal of Management Information Systems* 12.4 (1996). Copyright - Copyright M. E. Sharpe Inc. Spring 1996; Last updated - 2024-12-03, p. 5. URL: https://www-proquest-com.tudelft.idm.oclc.org/scholarly-journals/beyond-accuracy-what-data-quality-means-consumers/docview/218911948/se-2.

[9] Nicola Askham et al. “The six primary dimensions for data quality assessment”. In: *DAMA UK working group* (2013), pp. 432–435.

[10] Li Cai and Yangyong Zhu. “The Challenges of Data Quality and Data Quality Assessment in the Big Data Era”. In: *Data Science Journal* (May 2015). DOI: 10.5334/dsj-2015-002.

[11] Yuxin Wang, Joris Hulstijn, and Yao-Hua Tan. “Data quality assurance in international supply chains: an application of the value cycle approach to customs reporting”. In: *International Journal of Advanced Logistics* (June 2016), pp. 1–10. ISSN: 2287-7592. DOI: 10.1080/2287108x.2016.1178501.

[12] Carlo Batini et al. “Methodologies for data quality assessment and improvement”. In: *ACM Computing Surveys* 41.3 (July 2009), pp. 1–52. ISSN: 1557-7341. DOI: 10.1145/1541880.1541883.

[13] Donald P Ballou and Harold L Pazer. “Modeling data and process quality in multi-input, multi-output information systems”. In: *Management science* 31.2 (1985), pp. 150–162.

[14] Mats Bergdahl et al. “Handbook on data quality assessment methods and tools”. In: *Ehling, Manfred Körner, Thomas* (2007).

[15] Rick Verhulst. “Evaluating quality of event data within event logs: an extensible framework”. MA thesis. Eindhoven University of Technology: Eindhoven, The Netherlands, 2016.

[16] Jon Bokrantz et al. “Data quality problems in discrete event simulation of manufacturing operations”. In: *SIMULATION* 94.11 (Dec. 2017), pp. 1009–1025. ISSN: 1741-3133. DOI: 10.1177/0037549717742954.

[17] Yan Chen, Feibai Zhu, and Jay Lee. “Data quality evaluation and improvement for prognostic modeling using visual assessment based data partitioning method”. In: *Computers in Industry* 64.3 (Apr. 2013), pp. 214–225. ISSN: 0166-3615. DOI: 10.1016/j.compind.2012.10.005.

[18] Vincent S. de Feiter, Jessica M. I. Strickland, and Irene Garcia-Marti. “Advancing Data Quality Assurance with Machine Learning: A Case Study on Wind Vane Stalling Detection”. In: *Atmosphere* 16.2 (2025). ISSN: 2073-4433. DOI: 10.3390/atmos16020129. URL: https://www.mdpi.com/2073-4433/16/2/129.

[19] Shuo Jiang et al. “Prediction of vessel arrival time to port: a review of current studies”. In: *Maritime Policy & Management* (Apr. 2025), pp. 1–26. ISSN: 1464-5254. DOI: 10.1080/03088839.2025.2488376.

[20] United Nations Economic Commission for Europe (UNECE). *UN/LOCODE Code List by Country and Territory | UNECE*. Accessed: 2025-05-21. 2025. URL: https://unece.org/trade/cefact/unlocode-code-list-country-and-territory.

[21] Carlos Jahn and Tina Scheidweiler. “Port Call Optimization by Estimating Ships’ Time of Arrival”. In: *Dynamics in Logistics*. Ed. by Michael Freitag, Herbert Kotzab, and Jürgen Pannek. Cham: Springer International Publishing, 2018, pp. 172–177. ISBN: 978-3-319-74225-0.

[22] Sara El Mekkaoui, Loubna Benabbou, and Abdelaziz Berrado. “Machine Learning Models for Efficient Port Terminal Operations: Case of Vessels’ Arrival Times Prediction”. In: *IFAC-PapersOnLine* 55.10 (2022). 10th IFAC Conference on Manufacturing Modelling, Management and Control MIM 2022, pp. 3172–3177. ISSN: 2405-8963. DOI: https://doi.org/10.1016/j.ifacol.2022.10.217. URL: https://www.sciencedirect.com/science/article/pii/S2405896322022315.

[23] Paul Jumelet. “Commotie op het Suezkanaal: ongeluk ontsiert containerfeestje”. In: *Nieuwblad Transport* (June 2025). URL: https://www.nt.nl/scheepvaart/2025/06/23/commotie-op-het-suezkanaal-ongeluk-ontsiert-containerfeestje.

[24] Digital Container Shipping Association (DCSA). *Shipment Event Type Codes*. https://github.com/dcsaorg/DCSA-Information-Model/blob/master/datamodel/referencedata.d/shipmenteventtypecodes.csv. Accessed: 2025-05-07. 2025.

[25] Digital Container Shipping Association (DCSA). *Equipment Event Type Codes*. https://github.com/dcsaorg/DCSA-Information-Model/blob/master/datamodel/referencedata.d/equipmenteventtypecodes.csv. Accessed: 2025-05-07. 2025.

[26] Digital Container Shipping Association (DCSA). *Transport Event Type Codes*. https://github.com/dcsaorg/DCSA-Information-Model/blob/master/datamodel/referencedata.d/transporteventtypecodes.csv. Accessed: 2025-05-07. 2025.

A

Scientific Paper

Prediction of data quality within a DCSA-compliant container tracker

D. Hogendoorn[1], F. Schulte[1][0000−0003−3159−4393], N. Yorke-Smith[1][0000−0002−1814−3515], and B. van Riessen[2][0000−0003−2106−2617]

[1] Delft University of Technology, Delft 2628CD, The Netherlands
[2] Poort8 BV, Rotterdam, The Netherlands

Abstract. Accurate tracking of container shipments is vital for efficient global logistics operations, yet current event-based tracking systems face significant data quality (DQ) challenges. This research introduces a novel, systematic approach for assessing data quality within event-based container tracking systems compliant with the Digital Container Shipping Association (DCSA) standard. By employing supervised machine learning models (Linear Regression, Random Forest, XGBoost, and Neural Networks), the study predicts durations of container journeys and if the duration deviates at least one calender day from the duration provided, it classifies it as low-quality data. Results demonstrate that a relatively simple Linear Regression model significantly outperforms more complex models, highlighting the importance of domain-specific feature engineering. Furthermore, this research emphasizes that durations provided by carriers at early stages frequently exhibit inaccuracies. Consequently, the Linear Regression model offers substantial practical utility by reliably classifying early-stage durations as trustworthy or untrustworthy. Future research should integrate additional data sources, such as real-time AIS and environmental factors, to further enhance predictive robustness.

Keywords: Container tracking · DCSA · Machine-learning · data quality · ETA · Regression Analysis

1 Introduction

It has been estimated that about 90% of the world's trade is transported in cargo containers [5]. Accurate container tracking significantly influences logistics efficiency, scheduling reliability, and overall operational performance. As global trade volumes grow and supply chains become increasingly complex, the data infrastructures underpinning container tracking systems face heightened challenges. Even minor inaccuracies in container location, transit milestones, or estimated arrival times (ETAs) can escalate into considerable operational disruptions and financial consequences.

These challenges are amplified by the diverse range of data sources involved. Container tracking data originate from shipping lines, port authorities, terminal operators, and third-party logistics providers, each maintaining distinct data collection methods, standards, and reliability levels. This variability introduces inconsistencies, incomplete information, and inaccuracies, diminishing trust among stakeholders and complicating operational decisions.

The Digital Container Shipping Association (DCSA) developed the Track & Trace (T&T) standard to address these issues, providing a unified format for container transport event data. Although this standardization significantly enhances interoperability, it does not inherently guarantee the accuracy or completeness of the data exchanged.

This research addresses the urgent need for systematic data quality (DQ) assessment methods tailored specifically to DCSA-compliant event-based container tracking systems. By employing machine learning (ML) methods to predict deviations in ETAs, this study aims to proactively identify inaccuracies, ensuring timely, accurate, and trustworthy data for stakeholders.

2 Related literature

Data quality (DQ) assessment has been extensively explored, primarily through dimensional metrics such as accuracy, completeness, consistency, and timeliness. [12] provided foundational definitions for these metrics, with accuracy reflecting closeness to true values, completeness indicating data availability, consistency representing uniformity across databases, and timeliness measuring data currency. [1] refined these metrics into six core attributes to enhance practical applicability.

[8] emphasized the importance of dimensional metrics within supply chain analytics but highlighted their limitations when addressing complex, real-world DQ challenges. Similarly, [2] conducted a systematic review of methodologies for assessing and improving data quality across diverse sectors, categorizing processes and identifying associated costs. Their framework facilitates structured approaches for targeted quality enhancement.

While [4] recognized challenges in Big Data environments, proposing a comprehensive and contextualized DQ assessment framework, significant variability persists in how scholars categorize and prioritize these dimensions. [3] further underscored the variability in existing approaches by providing a structured implementation roadmap primarily tailored for statistical quality reporting, thus illustrating the lack of consensus.

In maritime logistics specifically, several studies addressed DQ through indirect or targeted approaches. [5] explored anomaly detection methods for maritime container itineraries to enhance customs and supply chain security, a useful but limited approach concerning broader event-based DQ. [6] presented visual assessment techniques for data readiness evaluations, while [7] utilized machine learning to identify specific data faults in other contexts, demonstrating potential applicability in container logistics.

[10] reviewed AIS-based studies on vessel arrival predictions, showcasing the successful use of advanced machine learning models such as Random Forest, XG-

Boost, and neural networks. [9] and [11] similarly utilized AIS data combined with environmental variables to optimize vessel arrival predictions at ports, underscoring the effectiveness of ML techniques.

Despite these advancements, there remains a notable gap in applying such comprehensive methods directly to DCSA-compliant event-based container tracking systems. Existing literature reveals no consensus or widely-adopted framework specifically for holistic DQ assessments within container tracking contexts. This lack of consensus leaves significant opportunities for developing and validating tailored methodologies that address the practical complexities of event-based container tracking data, thereby motivating this research.

3 Methodology

The methodology employed in this research consists of three primary stages. Initially, container tracking event data complying with the DCSA T&T standard were systematically collected and preprocessed. Preprocessing included consolidating event logs from multiple sources, removing duplicates, and segmenting container journeys into individual transport legs.

Next, a set of supervised machine learning models, namely Linear Regression (LR), Random Forest (RF), Extreme Gradient Boosting (XGBoost), and Neural Networks (NN), were trained to predict actual transport-leg durations. Training utilized a time-based train-test split to simulate real-world operational forecasting scenarios and avoid information leakage.

Lastly, predictions were assessed by comparing predicted arrival times to recorded ETAs. Data points were classified as low-quality if the predicted arrival deviated from the recorded ETA by more than one calendar day. Model performance was then evaluated using precision-oriented metrics designed to balance the costs of false alarms against missed detections, ensuring practical applicability within operational settings.

4 Results

The results demonstrate that the supervised machine learning models varied significantly in their predictive accuracy and their effectiveness in classifying data quality. The Linear Regression (LR) model, despite its computational simplicity, exhibited robust predictive performance with an $RMSE$ of 3.25 days and an R^2 of 0.381. In contrast, more complex models such as Random Forest (RF) and Neural Networks (NN) underperformed considerably. Specifically, the Random Forest (shown in Figure2) produced a notably worse $RMSE$ of 6.28 days and a negative R^2 value, indicating the model's inability to adequately fit the event data.

Similarly, the Neural Network model yielded an $RMSE$ of 4.20 days, underperforming compared to Linear Regression, with residuals highlighting substantial variability in predictions, as shown in Figure 3.

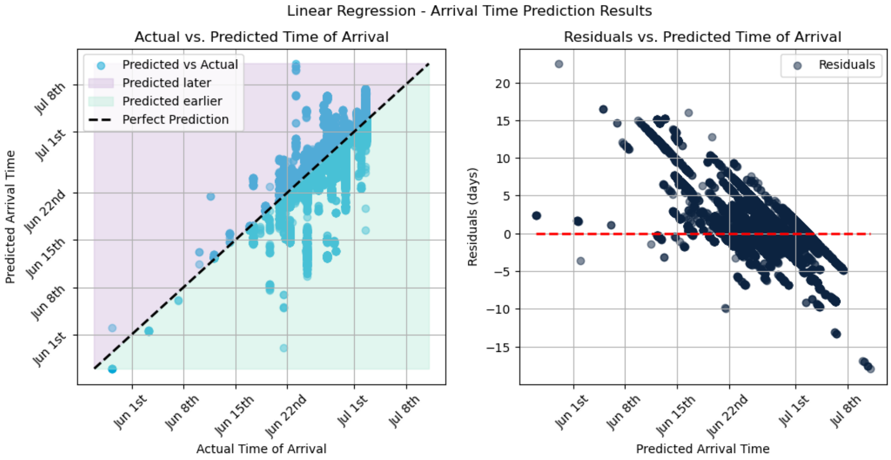


Fig. 1. Linear-regression performance using a time-based train–test split

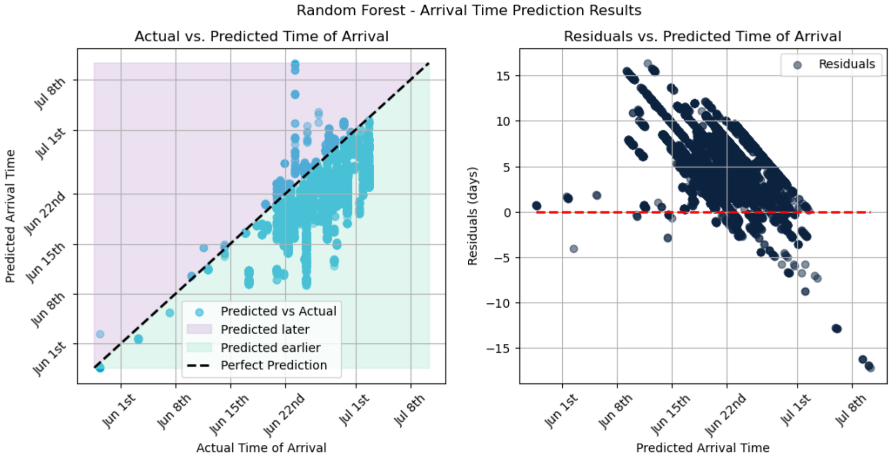


Fig. 2. Predicting durations with a Random Forest

Extreme Gradient Boosting (XGBoost) also demonstrated lackluster results with an $RMSE$ of 5.61 days, still notably behind the simpler LR approach. The results are shown in Figure 4.

The primary goal of this research was not merely accurate duration prediction but effective classification of event data quality. Here, Linear Regression also emerged as the most effective model, consistently achieving a precision-oriented F_β-score of approximately 68.5% at a β-value of 0.141. This β is specifically engineered so that a false positive has the same impact as having 50 false negatives. This score reflects an appropriate balance, heavily penalizing false alarms

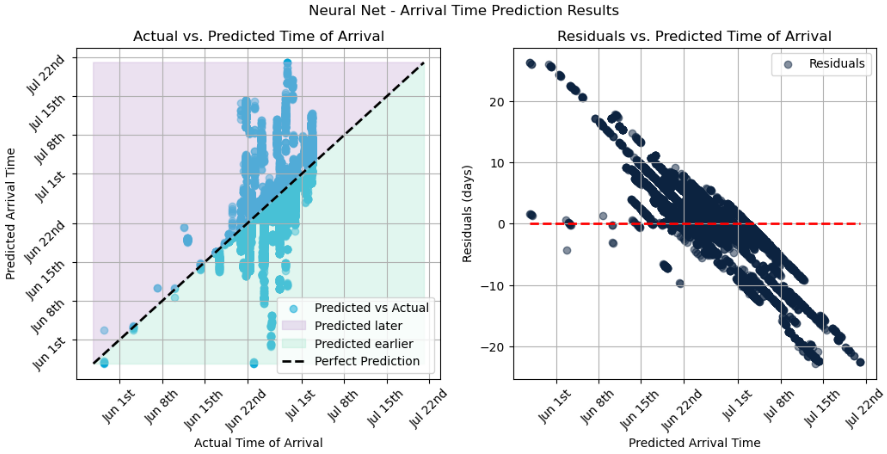


Fig. 3. Neural-network predictions of leg duration

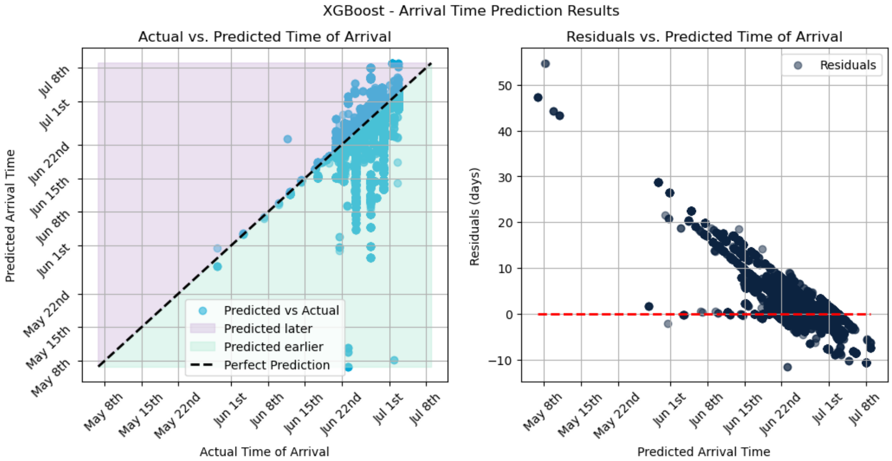


Fig. 4. Predicting durations with XGBoost

while tolerating occasional missed detections, aligned with operational priorities defined by industry stakeholders. XGBoost achieved a comparable F-score of 56.8%, whereas Random Forest and Neural Networks lagged further behind, primarily due to their excessive generation of false alarms.

Since ETAs provided by terminals are usually more accurate than ETAs provided by carriers, and ETAs get more and more accurate as a container approaches its destination, the value of this research is especially relevant for classifying ETAs that are published at most a week before ATA and only for ETAs provided by carriers. The performance of the models improved further when analyses were restricted exclusively to early-stage ETAs provided by carriers, reinforcing the notion that carrier-generated ETAs in early stages of transport are generally less reliable. Under this scenario, Linear Regression exhibited increased classification accuracy with fewer false alarms and missed detections, yielding an F-score of 72.0%, further validating its suitability in practical, carrier-oriented operational contexts.

A summary of the classification performance of all 4 models is provided in Table 1.

Table 1. $\mathbf{F_{0.141}}$ score for all models

Model	LR	RF	XGBoost	NN
Overall model performance	68.5%	49.3%	56.8%	50.0%
Early carrier classification performance	72.0%	64.7%	71.6%	64.2%

5 Discussion

Several limitations warrant consideration. The dataset's relatively short temporal scope, coupled with the inherent limitations of relying exclusively on event-based data without incorporating real-time AIS or environmental data, may restrict the models' broader applicability. Future research could therefore benefit from integrating additional external data sources, such as AIS trajectories and real-time environmental conditions (e.g., weather disruptions or port congestion), potentially enhancing predictive robustness and adaptability.

Additionally, given the observed variability in carrier-provided ETAs, further research could explore meta-learning techniques to dynamically adapt to operational shifts or disruptions, thereby maintaining high predictive accuracy despite evolving operational conditions.

6 Conclusion

This research provides a systematic approach to assessing data quality within DCSA-compliant event-based container tracking systems. Findings demonstrate that straightforward predictive models, particularly Linear Regression, effectively identify inaccuracies in ETAs, offering robust performance and operational practicality. The study confirms that rigorous preprocessing and domain-specific feature engineering significantly influence model outcomes.

The practical value of this research is particularly relevant for early-stage ETAs from carriers, which frequently exhibit significant deviations. Future research should address existing limitations by incorporating broader datasets and external variables to enhance the generalizability and resilience of predictive models, ultimately contributing to more reliable global container logistics operations.

Acknowledgments. All data used in this study was kindly provided by the container tracker software *HeyWim Container Tracking*, which is a product of Poort8 BV.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Askham, N., Cook, D., Doyle, M., Fereday, H., Gibson, M., Landbeck, U., Lee, R., Maynard, C., Palmer, G., Schwarzenbach, J.: The six primary dimensions for data quality assessment. DAMA UK working group pp. 432–435 (2013)
2. Batini, C., Cappiello, C., Francalanci, C., Maurino, A.: Methodologies for data quality assessment and improvement. ACM Computing Surveys **41**(3), 1–52 (Jul 2009). https://doi.org/10.1145/1541880.1541883
3. Bergdahl, M., Ehling, M., Elvers, E., Földesi, E., Körner, T., Kron, A., Lohauß, P., Mag, K., Morais, V., Nimmergut, A., et al.: Handbook on data quality assessment methods and tools. Ehling, Manfred Körner, Thomas (2007)
4. Cai, L., Zhu, Y.: The challenges of data quality and data quality assessment in the big data era. Data Science Journal (5 2015). https://doi.org/10.5334/dsj-2015-002
5. Camossi, E., Dimitrova, T., Tsois, A.: Detecting anomalous maritime container itineraries for anti-fraud and supply chain security. In: 2012 European Intelligence and Security Informatics Conference. pp. 76–83. IEEE (Aug 2012). https://doi.org/10.1109/eisic.2012.39
6. Chen, Y., Zhu, F., Lee, J.: Data quality evaluation and improvement for prognostic modeling using visual assessment based data partitioning method. Computers in Industry **64**(3), 214–225 (Apr 2013). https://doi.org/10.1016/j.compind.2012.10.005
7. de Feiter, V.S., Strickland, J.M.I., Garcia-Marti, I.: Advancing data quality assurance with machine learning: A case study on wind vane stalling detection. Atmosphere **16**(2) (2025). https://doi.org/10.3390/atmos16020129, https://www.mdpi.com/2073-4433/16/2/129
8. Hazen, B.T., Boone, C.A., Ezell, J.D., Jones-Farmer, L.A.: Data quality for data science, predictive analytics, and big data in supply chain management: An introduction to the problem and suggestions for research and applications. International Journal of Production Economics **154**, 72–80 (Aug 2014). https://doi.org/10.1016/j.ijpe.2014.04.018
9. Jahn, C., Scheidweiler, T.: Port call optimization by estimating ships' time of arrival. In: Freitag, M., Kotzab, H., Pannek, J. (eds.) Dynamics in Logistics. pp. 172–177. Springer International Publishing, Cham (2018)
10. Jiang, S., Liu, L., Peng, P., Xu, M., Yan, R.: Prediction of vessel arrival time to port: a review of current studies. Maritime Policy & Management pp. 1–26 (Apr 2025). https://doi.org/10.1080/03088839.2025.2488376
11. Mekkaoui, S.E., Benabbou, L., Berrado, A.: Machine learning models for efficient port terminal operations: Case of vessels' arrival times prediction. IFAC-PapersOnLine **55**(10), 3172–3177 (2022). https://doi.org/https://doi.org/10.1016/j.ifacol.2022.10.217, https://www.sciencedirect.com/science/article/pii/S2405896322022315, 10th IFAC Conference on Manufacturing Modelling, Management and Control MIM 2022

12. Wang, R.Y., Strong, D.M.: Beyond accuracy: What data quality means to data consumers: Jmis. Journal of Management Information Systems **12**(4), 5 (1996), https://www-proquest-com.tudelft.idm.oclc.org/scholarly-journals/beyond-accuracy-what-data-quality-means-consumers/docview/218911948/se-2, copyright - Copyright M. E. Sharpe Inc. Spring 1996; Last updated - 2024-12-03

B

DCSA events

Tables B.1, B.2, and B.3 provides an overview of all the events used in the DCSA standard.

Table B.1: Codes used for shipment events in the DCSA standard. Retrieved from [24]

Event Code	Event Name	Event Description
RECE	Received	Indicates that a document is received by the carrier or shipper
DRFT	Drafted	Indicates that a document is in draft mode being updated by either the shipper or the carrier.
PENA	Pending Approval	Indicates that a document has been submitted by the carrier and is now awaiting approval by the shipper.
PENU	Pending Update	Indicates that the carrier requested an update from the shipper which is not received yet.
PENC	Pending Confirmation	Indicates that a document has been submitted by the shipper and is now awaiting approval by the carrier.
REJE	Rejected	Indicates that a document has been rejected by the carrier.
APPR	Approved	Indicates that a document has been approved by the counterpart.
ISSU	Issued	Indicates that a document has been issued by the carrier.
SURR	Surrendered	Indicates that a document has been surrendered by the customer to the carrier.
SUBM	Submitted	Indicates that a document has been submitted by the customer to the carrier.
VOID	Void	Cancellation of an original document.
CONF	Confirmed	Indicates that the document is confirmed.
REQS	Requested	A status indicator that can be used with a number of identifiers to denote that a certain activity, service or document has been requested by the carrier, customer or authorities. This status remains constant until the requested activity is “Completed”.
CMPL	Completed	A status indicator that can be used with a number of activity identifiers to denote that a certain activity, service or document has been completed.
HOLD	On Hold	A status indicator that can be used with a number of activity identifiers to denote that a container or shipment has been placed on hold i.e. can’t progress in the process.
RELS	Released	A status indicator that can be used with a number of activity identifiers to denote that a container or shipment has been released i.e. allowed to move from depot or terminal by authorities or service provider.
CANC	Cancelled	A status indicator to be used when the booking is cancelled by the Shipper

Table B.2: Codes used for equipment events in the DCSA standard. Retrieved from [25]

Event Code	Event Name	Event Description
LOAD	Load	The action of lifting cargo or a container on board of the mode of transportation. Load is complete once the cargo or container has been lifted on board the mode of transport and secured.
DISC	Discharge	The action of lifting cargo or containers off a mode of transport. Discharge is the opposite of load.
GTIN	Gate in	The action when a container is introduced into a controlled area like a port - or inland terminal. Gate in has been completed once the operator of the area is legally in possession of the container.
GTOT	Gate out	The action when a container is removed from a controlled area like a port – or inland terminal. Gate-out has been completed once the possession of the container has been transferred from the operator of the terminal to the entity who is picking up the container.
STUF	Stuffing	The process of loading the cargo in a container or in/onto another piece of equipment.
STRP	Stripping	The action of unloading cargo from cantainers or equipment.
PICK	Pick-up	The action of collecting the container at customer location.
DROP	Drop-off	The action of delivering the container at customer location.
INSP	Inspected	Identifies that the seal on equipment has been inspected.
RSEA	Resealed	Identifies that the equipment has been resealed after inspection.
RMVD	Removed	Identifies that a Seal has been removed from the equipment for inspection.
AVPU	Available for Pick-up	Identifies that shipment/ Container is ready to be picked up / collection at a facility.
AVDO	Available for Drop-off	Identifies that shipment/ container is ready to be dropped off / delivered at a facility
CUSS	Customs Selected for Scan	Identifies that Customs has selected the equipment for scanning
CUSI	Customs Selected for Inspection	Identifies that that Customs has selected the equipment for inspection
CUSR	Customs Released	Identifies that Customs has released the equipment for either export from or import into the country.
WAYP	Way Point Crossed	A waypoint is an intermediate point or place during transit of shipment, waypoint crossed indicates that the equipment has crossed the particular waypoint on its transit.

Table B.3: Codes used for transport events in the DCSA standard. Retrieved from [26]

Event Code	Event Name	Event Description
ARRI	Arrival	
DEPA	Departure	

C

UN/LO Codes mapping

In Table C.1 the mapping of incorrect UNLO codes to correct UNLO codes can be found.

Table C.1: Mapping of incorrect UN/LO to correct UN/LO codes

Incorrect UN/LO	Correct UN/LO
NLVNL	NLVEN
NLALP	NLAPN
NLTIL	NLTLB
NLNRN	NLNIJ
NLNRJ	NLNIJ
NLVL	NLVEN
NLTI	NLTLB
NLNRD	NLNIJ
NLNRB	NLNIJ
NLNR	NLNIJ
NLVLX	NLVEN
TNTNG	MAPTM
MAMED	MAPTM
KRBUS	KRBNP
BGCGP	BDCGP
PKBIN	PKBQM
MAMSA	MAPTM
ECPOS	ECPSJ
MAMDI	MAPTM
PKMBQ	PKBQM
ILASD	ILASH
HRRIJ	HRRJK
EGDMT	EGDAM
PKQAS	PKBQM
MAMTD	MAPTM
ILHAI	ILHFA
MAMES	MAPTM
MAMON	MAPTM

D

Feature Weights

Table D.1 reports the absolute coefficients of the linear-regression model, rounded to two decimal places. After one-hot encoding, the feature matrix contains 491 columns: 459 OD-pair dummies, 12 carrier dummies, and 16 ETA-publisher dummies, in addition to four numeric variables. Because listing all dummy coefficients is impractical, the table shows the mean absolute value for each categorical group; absolute values are used to avoid the misleading cancellation of positive and negative effects. A large mean absolute coefficient indicates that the corresponding group exerts substantial leverage on the prediction surface, whereas values near zero have negligible influence.

Table D.1: Weight of features

Feature	Weight
Published ETA	874,788.64
OD pair (mean of 459 dummies)	216,665.62
ETA publisher (mean of 16 dummies)	150,490.55
Carrier (mean of 12 dummies)	140,658.42
Time before ETA	83,714.44
total days tracked	17,041.59
Departure timestamp	3,423.22