ALMA MATER STUDIORUM – UNIVERSITÀ DI BOLOGNA
CESENA CAMPUS

Department of Computer Science and Engineering - DISI
Second-cycle Degree in Digital Transformation Management
Class: LM-91

Predicting Trial Class Attendance in EdTech: A Machine Learning Approach to Funnel Optimization

Graduation thesis in
Machine learning and data mining

Supervisor
Prof. Alina Sîrbu

Candidate
Alina Iakubova

Controrelatore
Prof. Matteo Golfarelli

(1-st) Session
Academic Year 2025-2026

Acknowledgements

I would like to express my sincere gratitude to my supervisor, Prof. Alina Sîrbu, for her valuable guidance and continuous support throughout the development of this thesis.

Contents

List of Figures

Listings

Chapter 1

Introduction

Digital transformation in customer-facing industries is increasingly defined by the capacity of organizations to translate behavioural data into operational decisions[45]. Core acquisition departments, such as sales and marketing, no longer can lead only on conversion rates: companies need to understand, at the very granular user-level, who is likely to become a subscriber, who is likely to drop off, and how internal resources should be allocated to maximize return on acquisition spending. Within this shift, predictive analytics has moved from being a solely technical instrument to becoming a crucial component of the infrastructure of operational processes that supports day-to-day managerial choices[44].

This thesis is at the intersection of data analytics, machine learning, and digital transformation, with a specific focus on lead scoring and trial-class attendance prediction in the EdTech sector. This work is the result of an applied research project carried out during an extracurricular internship at Kodland PTE. LTD in analytics department. The internship allowed to acquire an understanding of the entire customer acquisition process and the data systems that support it. Based on these empirical data, the thesis examines whether a predictive machine learning model can improve the efficiency of lead qualification in a real acquisition funnel, what theoretical and methodological basis is needed for the practical application of the model, how to deploy and integrate the model into existing business processes, and which managerial actions follow these processes.

The Educational Technology (EdTech) sector has experienced rapid growth over the past decade, accelerated by the global shift to remote learning and by sustained investment in digital education platforms after the Covid pandemic[47]. This thesis focuses on educational technology companies offering remote lessons for children and teenagers. These companies rely heavily on digital marketing as their primary customer acquisition channel which includes paid social campaigns in Google and Meta, search advertising, and customised landing pages. They generate large volumes of leads, but also bring significant customer acquisition costs that must be covered by downstream conversion. One of the core characteristics of attracting a new client in Edtech sector

is the free trial class which serves as central conversion event in the customer journey. The trial class is the moment when the educational value proposition is demonstrated, allowing the client to decide whether they are ready to make a purchase. Consequently, the economics of acquisition depend not on the generated volume of leads, but on those who actually attend the trial class[33].

Customer acquisition in EdTech is therefore characterized by three interrelated challenges: lead heterogeneity, asymmetry between bookings and behavioural intent and resource constraints. First one includes prospective customers who reach the booking stage differ widely in motivation. Second one is that a booking is a weak signal often produced by curiosity, while attendance is a stronger signal of accrual interest. And the last one is that tutors and sales managers cannot scale linearly with lead volume.

To address these challenges, it is necessary to move from descriptive analytics to predictive approaches that can identify potential customers in advance and determine the optimal allocation of resources at subsequent stages.

The main objective of this thesis is to design and evaluate a machine learning–based lead scoring model for predicting trial-class attendance in an EdTech acquisition funnel, using the Kodland internship as the empirical setting. The general objective is articulated into three more specific ones: conceptual, methodological and data-driven. Conceptual consists of review lead qualification and attendance prediction researches. Methodological includes a formulation of the prediction task to construct a feature set that is available at the moment of scoring, with attention to the risk of data leakage. Data driven is the translation of model outputs into actionable insights and recommend how they can be integrated into existing business processes.

The thesis makes the following contributions:

- **Definition of EdTech lead scoring.** A structured description of how customer acquisition in the Edtech differs from more typical e-commerce scenarios, with a particular focus on trial lesson.
- **Machine-learning model and feature examination** A predictive model is developed and evaluated, behavioral features are investigated to differentiate user by their motivation.
- **Empirical evaluation on a real EdTech funnel.** Validation results showing a significant behavioural difference across score buckets, with attendance rates rising from 8.1% in the low-probability segment to 19.0% in the medium-probability segment and 36.5% in the high-probability segment, against a baseline of approximately 7–8%.
- **Interpretable, business-usable output.** A three-bucket segmentation (low, medium, high) that can be directly used by non-technical stakeholders for prioritization, follow-up logic, and resource allocation.
- **Reflection on digital transformation in mid-size EdTech operations.** A discussion of how predictive analytics interacts with broader digital transformation

processes is conducted.

To support transparency and partial reproducibility of these results, all the code developed for this study, including the pre-processing step, the machine learning scripts, and the post-hoc analysis tools, has been made publicly available on GitHub at `https://github.com/AlinaYakubova/Thesis_DTM_Trial_Class_Prediction`. The underlying dataset cannot be released due to a non-disclosure agreement with the host company, and all results reported in this thesis are based on anonymized, aggregated outputs.

The remainder of this thesis is organized as follows. Chapter 2 provides the empirical basis of the work, introducing Kodland and the scope of the internship. Chapter 3 reviews the relevant literature on customer acquisition and predictive analytics in marketing. Chapter 4 presents the theoretical background, including the machine learning task and the evaluation metrics. Chapter 5 describes the available data sources and the exploratory analyses conducted on them. Chapter 6 details the methodology, including dataset construction, feature engineering, leakage handling, and validation design. Chapter 7 presents and discusses the results. Chapter 8 concludes with operational integration, limitations, and ethical considerations. Chapter 9 describes the main conclusions of the study and provides recommendations for future work.

Chapter 2

Business Context: Kodland and the Internship Scope

2.1 Company Overview

Kodland is an EdTech company operating in the online coding education segment, offering programming courses for children and teenagers. The company relies predominantly on performance marketing to attract prospective students, directing paid traffic from digital advertising campaigns to dedicated landing pages. Its business model is trial-and-subscription-based: the primary conversion objective is to move a prospective user from initial advertisement banner through a acquisition funnel toward a booked and attended trial class, after which offer and purchase become the target outcome.

The global EdTech market has experienced substantial growth over the past decade. Within this landscape, coding education for children occupies a fast-growing niche, driven by increasing parental demand for digital literacy skills and growing institutional interest in computational thinking [48]. Kodland operates across multiple geographic markets, with a notable expansion into the Latin American (LatAm) segment during the period covered by this thesis. This market presents specific operational characteristics, including lower baseline data availability, distinct user behaviour patterns, and limited deployment infrastructure that shaped the analytical choices described in subsequent chapters and represent a key boundary condition for the generalisability of the modelling approach.

2.2 The Acquisition Funnel and LatAm Specifics

Kodland's acquisition process follows a structured digital funnel comprising several sequential stages [12]. A user's journey typically begins with exposure to a paid advertisement: banner, video, or social media creative, which redirects to a landing page.

Depending on the market and campaign type, landing pages may take the form of simple registration forms or multi-step interactive flows. Upon completing the landing page interaction, a user is classified as a lead and enters the customer relationship management (CRM) pipeline, where follow-up activities are managed by sales and operations teams. Operations teams are responsible for next communication, usually with reminder of upcoming trial class.

Each transition between stages represents a measurable drop-off point, and funnel efficiency, defined as the rate at which leads progress through successive stages, is a core operational metric [26]. This thesis focuses specifically on the booking-to-attendance transition, which emerged during the internship as the most operationally significant and analytically tractable conversion gap.

A distinctive feature of Kodland's acquisition process, particularly in the LatAm segment, is the use of quiz-based landing pages. Unlike a passive registration form, the quiz is designed to capture parental motivation and intent at the earliest point of funnel entry, consistent with the progressive profiling approach documented in the digital marketing literature [25].

The internship included one of the first systematic attempts to design and deploy a qualification-oriented scoring quiz specifically for the LatAm market. This required balancing two competing considerations: the informational value of quiz responses as predictive features, and the risk of response fatigue or abandonment if the flow was perceived as overly intrusive [17]. The resulting quiz design represented an operational compromise between qualification depth and conversion rate between questions. In figure 2.1, design of pages and some quiz questions are illustrated. An additional signal

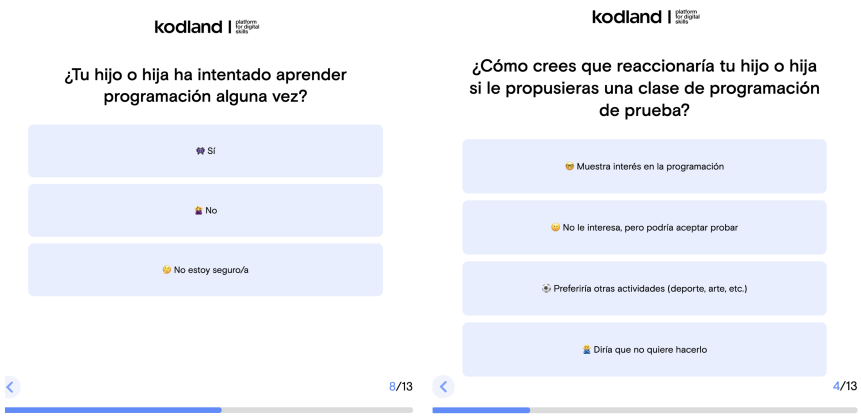

Figure 2.1: Kodland Latam Quiz Example

explored during the internship was the behavioural trace data generated by the quiz interaction itself: for example, the day and time when the quiz was completed, speed

of the quiz completion. These signals were hypothesised to correlate with user intent and subsequent attendance behaviour, and were evaluated as candidate features in the predictive model.

2.2.1 Trial Class as a Critical Funnel Step

The trial class is the most important step in Kodland's acquisition process. It functions as the primary conversion mechanism: a free introductory session for parent and their child, designed to demonstrate product value and facilitate the transition from lead to paying customer. From an operational point of view, the trial class requires the allocation of tutor time, scheduling resources, and follow-up capacity, making it a non-trivial investment per lead. Tutor time is a resource that Kodland pays for in all scenarios, regardless of whether the lead attended the lesson or not. Because the trial class is offered at no cost to the user, booking rates tend to be relatively high. However, the low commitment cost also contributes to elevated no-show rates: users who book a trial class are not contractually obligated to attend, and a meaningful proportion do not. This creates an asymmetry between the cost faced by the company and the uncertainty regarding whether those resources will be utilised. This dynamic is consistent with the broader literature on free trial conversion in subscription-based businesses, where the gap between trial initiation and active engagement is a well-documented challenge [13].

The gap between trial class bookings and actual attendance is the central operational problem addressed by this thesis. Analysis of available funnel data confirmed that a meaningful proportion of booked leads did not attend their scheduled trial class, a pattern consistent across markets and campaign types. Thus, a predictive scoring model that estimates attendance probability after the event of booking would allow the operations team to differentiate follow-up intensity and reallocate effort toward higher-probability leads. This would be considered as an application of operationally embedded machine learning consistent with recent frameworks for AI-driven marketing operations [14].

2.2.2 Current Lead Qualification Process

Prior to the introduction of the predictive scoring model, lead qualification at Kodland relied primarily on human interaction: sales and operations staff assessed lead quality through phone calls, follow-up messages, and informal judgement developed through experience. In practice, this meant that a tutor or sales representative would contact a booked lead, attempt to confirm attendance, and form a qualitative assessment of likelihood to show. No structured scoring mechanism existed to rank incoming leads by predicted conversion probability at the point of booking.

This approach had three systematic limitations. First, it was not scalable: as lead volumes increased, the manual qualification load grew proportionally, reducing per-lead attention and increasing response latency. Second, it was inconsistent: assessments

varied across team members and were influenced by individual heuristics rather than standardised criteria. Third, and most importantly for this thesis, it was reactive: qualification judgements were made after booking, rather than at an earlier funnel stage where prioritisation process would have greater impact. These limitations are consistent with documented shortcomings of rule-based and manual scoring systems in high-volume B2C acquisition environments [6].

2.3 Data Infrastructure

The analytical infrastructure available during the internship included three primary data source categories. First one, web tracking data, collected subject to cookie consent, captured device information and sessions parameters for users entering the Kodland website. Second, quiz questions and answers for them provided structured qualification signals from landing page interactions, available primarily for leads entering through quiz-enabled flows in the LatAm segment. These questions included motivational aspects of the kid, motivational aspects of the parent and general information about internet connection speed in the household. The last one, CRM signals, were generated through the sales pipeline, recorded booking events (reschedules) and attendance outcomes.

These sources were stored centrally in Amazon Redshift database.However, their completeness and consistency varied: web tracking data were subject to consent-based gaps and cross-device attribution limitations, quiz response data showed anomalies caused by frontend data collection issues and subsequent transformations in Redshift.

The technical environment for the project consisted of Amazon Redshift as the central data warehouse, Python for data preprocessing, feature engineering, and model development, Amazon QuickSight for dashboard development and reporting, and GitLab for version control and workflow management. SQL was used extensively for data extraction and building a table for quick download through Python. This stack is broadly representative of a mid-scale EdTech analytics environment, combining cloud-based storage and visualisation infrastructure with scripted analytical workflows [27].

2.4 Operational Problem Statement

The operational problem addressed by this thesis can be stated as follows: given the information available at the point of trial class booking, including web tracking signals, quiz responses, and CRM metadata, is it possible to estimate, with sufficient accuracy and operational reliability, the probability that a given lead will attend the booked trial class? This question has both a technical dimension, concerning model architecture, feature construction, and validation methodology, and an operational dimension, concerning deployment feasibility, interpretability requirements, and integration with

existing workflows.

Several contextual constraints shaped the scope of the problem. The LatAm segment with the deployed quiz was characterised by limited historical data, reducing the volume available for training and validation. Real-time Google Analytics data were not available as model inputs, constraining the feature set to CRM signals, quiz responses, and Google Analytics events that happened at least 1 day in advance of quiz completion. Deployment feasibility was conditioned by the existing stack: model output needed to be integrable into Redshift-based pipelines and surfaced through QuickSight dashboards without requiring bespoke infrastructure.

These constraints are treated throughout the thesis not as obstacles but as defining characteristics of the research setting: one that is representative of the operational realities faced by growth-stage EdTech companies attempting to introduce machine learning into marketing and conversion workflows [38].

Chapter 3

Literature Review

3.1 Digital Transformation in EdTech and Marketing

The concept of digital transformation has been extensively theorised across organisational and managerial disciplines, broadly defined as the process through which organisations integrate digital technologies into all areas of their operations, fundamentally altering how they deliver value and make decisions [46]. In the context of education, digital transformation has accelerated substantially over the past decade, driven by the evolution of online learning platforms, the commoditisation of cloud infrastructure, and the shift in learner expectations toward on-demand, personalised experiences [48].

Within the EdTech sector specifically, digital transformation has considered not only product delivery: the migration from physical to digital environments, but also in the marketing and acquisition functions that drive the expansion of these businesses. Performance marketing, data-driven funnel management, and algorithmic lead qualification represent a convergence of digital marketing practice and operational analytics that is characteristic of mature EdTech organisations [25]. The adoption of CRM systems, marketing automation platforms, and data warehousing infrastructure has enabled EdTech companies to collect and act on granular behavioural signals across the user journey, from first advertisement exposure to long-term retention [26].

However, the academic literature on digital transformation in EdTech has concentrated disproportionately on pedagogical outcomes and learning technology adoption, paying less attention to the marketing and acquisition operations that underpin commercial viability [43]. This thesis contributes to closing that gap by examining how predictive analytics and machine learning, when embedded in acquisition workflows, constitute a form of operational digital transformation - one in which algorithmic decision support replaces or augments manual qualification processes.

3.2 The No-Show Problem Across Domains

The phenomenon of scheduled appointment non-attendance, colloquially referred to as the no-show problem, has been studied across multiple applied domains. In each context, no-show behaviour generates asymmetric costs: the service provider incurs preparation and opportunity costs regardless of whether the client appears, while the client faces no material consequence for non-attendance when the service is free or low-commitment.

3.2.1 Healthcare

The healthcare domain provides the most developed literature on no-show prediction, driven by the substantial operational and financial costs of missed medical appointments. Studies consistently identify a robust set of predictors across settings: gap time between booking and appointment date, prior no-show history, patient age, appointment type, and scheduling channel [10]. Machine learning approaches, including logistic regression, random forests, and gradient boosting, have been applied to predict missed appointments with AUC values typically ranging from 0.70 to 0.82 across the published literature [10]. A recurring finding is that a gap time is among the strongest individual predictors: the longer the interval between booking and appointment, the higher the probability of non-attendance, a pattern attributed to the decay of initial motivation over time [36].

Importantly, the healthcare literature has also captured with the operational integration of predictive models, specifically, how predicted no-show probabilities should inform overbooking, reminder scheduling, and outreach prioritisation. This translational challenge, bridging model output and operational action, is directly analogous to the deployment context addressed in this thesis.

3.2.2 Other Domains

Beyond healthcare, no-show behaviour has been studied in restaurant reservations [2], airline and transportation management [22], and online scheduling contexts [39]. Across these domains, a common structural pattern emerges: non-attendance is more likely when the commitment cost at the point of booking is low, when the time between booking and event is long, and when the service is perceived as easily reschedulable or replaceable. These conditions apply directly to the EdTech trial class context, where the free nature of the trial and the availability of alternative timeslots reduces the psychological cost of non-attendance.

The literature on restaurant reservations has contributed methodological insights relevant to this thesis. Specifically, [2] analysed the strategic role of reservations in capacity-constrained services, yielding findings about customer behavior that have direct analogues in the patterns captured by CRM and web tracking signals within a typical EdTech acquisition pipeline. In transportation and service operations, overbooking

strategies have long been predicated on empirical estimates of passenger or customer non-show rates, with service pricing tier and booking-to-service lead time recognised as systematic structural determinants of non-attendance probability [22]. In the online scheduling context, [39] demonstrated that cultural and temporal factors significantly affect how users engage with scheduled commitments, a finding particularly relevant in multi-market EdTech deployments spanning different geographies and time zones.

3.3 Common Predictive Patterns: Behavioural Signals, Gap Time, and Demographics

Across domains, the literature converges on three categories of predictors that reliably differentiate between attenders and non-attenders.

Behavioural signals are actions taken by the user prior to the scheduled event. They are consistently among the strongest predictors. In healthcare, prior attendance history is the single most predictive variable [10]. In digital marketing contexts, engagement signals such as email open rates, click-through behaviour, and session depth have been shown to correlate with downstream conversion [34]. For the EdTech acquisition funnel, behavioural signals available at or before the booking point include quiz completion behaviour, response speed, time of day of interaction, and source channel, all of which constitute candidate features in the modelling work reported in this thesis.

Gap time is the interval between the booking event and the scheduled appointment. It shows a consistent negative effect on attendance probability across all studied domains [36]. This effect is theorised through the lens of temporal discounting: as the gap between commitment and fulfilment increases, the motivation of the original decision decreases [36]. In the EdTech context, trial class booking-to-attendance gap time varies across markets and campaign types, making it a candidate feature of particular theoretical and practical relevance for the predictive model.

Demographic and contextual factors include age, location, device type, and time of booking. They have been shown to moderate no-show rates in multiple settings [10]. In LatAm market contexts specifically, demographic and contextual signals captured through the qualification quiz represent a potentially distinctive source of predictive information, as they represent parental motivation and household characteristics that web tracking data alone cannot capture. The interaction between demographic context and behavioural engagement signals has received growing attention in multichannel customer management research [34], reinforcing the relevance of both signal types as complementary rather than substitutable predictors.

3.4 Machine Learning for Conversion and Engagement Prediction

The application of machine learning approaches to conversion and engagement prediction in marketing contexts has expanded substantially over the past decade, driven by improvements in data infrastructure and the commoditisation of ML tooling [38]. Lead scoring can be defined as a task of ranking prospective customers by their predicted likelihood of conversion. This has been a primary application area, with both academic and practical contributions documenting the superiority of data-driven approaches over rule-based systems in high-volume acquisition environments [6].

Logistic regression remains a widely used baseline in lead scoring applications due to its interpretability and calibration properties [21]. Ensemble methods, particularly gradient boosting algorithms such as XGBoost and LightGBM, have demonstrated consistent performance improvements over logistic regression on structured tabular data [11], though at the cost of reduced interpretability. For operational deployment contexts where model outputs must be understood and acted upon by non-technical stakeholders, the trade-off between predictive performance and interpretability is a central design consideration [40].

Class imbalance, a structural feature of conversion prediction tasks where positive outcomes are substantially rarer than negative outcomes, is a recurring methodological challenge. Techniques including SMOTE oversampling, cost-sensitive learning, and threshold optimisation have been applied to address this challenge, with empirical results suggesting that threshold calibration to operational objectives is often more impactful than resampling per se [19]. This finding is directly relevant to the modelling approach adopted in this thesis, where the primary operational objective was not accuracy maximisation but the maximal identification of the attendance segment.

Beyond model selection and class imbalance handling, the literature has increasingly emphasised the importance of model interpretability through post-hoc explanation methods. SHAP (SHapley Additive exPlanations) values, introduced by [30], that provide a theoretically grounded framework for decomposing individual predictions into per-feature contributions, enabling analysts to identify which input variables drove each prediction. In operational lead scoring contexts, SHAP analysis serves a dual purpose: it supports model validation by confirming that predictions are driven by substantively meaningful features rather than data artefacts, and it provides actionable insight to business teams who must act on model outputs [40]. The use of SHAP analysis in this thesis is consistent with current best practice in applied machine learning for business decision-making.

3.4.1 Machine Learning Integration and Human-in-the-Loop Decision Systems

The gap between model development and operational deployment is a recognised challenge in applied machine learning [42]. Academic treatments of ML for conversion prediction frequently evaluate models in offline, retrospective settings, without addressing the practical requirements of integrating model outputs into live business processes. The deployment approach adopted in this thesis is a batch scoring through SQL pipeline with output through a BI dashboard, is consistent with this pragmatic model and reflects the organisational constraints common to growth-stage technology companies.

A distinct but related literature addresses the design of decision systems in which algorithmic outputs inform but do not replace human judgement are called human-in-the-loop (HITL) systems [32]. In marketing operations, HITL architectures are particularly relevant when model outputs have direct consequences for resource allocation, and where the cost of false positives and false negatives is asymmetric. The literature identifies several design principles for effective HITL systems: outputs should be interpretable to non-technical operators, uncertainty should be communicated rather than suppressed, and the system should support rather than automate final decisions [9].

The scoring model developed in the context of this thesis was designed with these principles: rather than producing a single binary classification, it generated a probability score bucketed into three operational levels (low, medium, high), allowing sales and operations teams to apply their own judgement about follow-up intensity within each tier. This design choice reflects the broader argument that operational ML systems in marketing contexts are most effective when they function as decision-support tools rather than decision-replacement systems [32].

3.5 Summary and Research Gap

The reviewed literature establishes several points relevant to this thesis. First, no-show behaviour is a well-documented and costly operational problem across multiple service domains, with predictive modelling having demonstrated consistent practical value particularly in healthcare [10]. Second, a common set of predictive signals: gap time, prior behavioural engagement, and demographic context, generalises across domains, suggesting that patterns identified in non-EdTech settings are transferable to the trial class attendance prediction task [36]. Third, ML-based lead scoring has been shown to outperform rule-based approaches in high-volume B2C acquisition environments, though the operational integration of such models remains an underexplored area [6]. Fourth, the HITL literatures provide practical frameworks for deploying ML outputs in operational contexts, emphasising interpretability, calibration, and human oversight [32].

However, several gaps in the existing literature are directly addressed by this thesis. The no-show prediction literature is dominated by healthcare applications; applications to EdTech trial class attendance are mostly absent from the published record. The role of qualification quiz responses, particularly those designed to capture parental motivation, as predictive features for attendance behaviour has not been studied in the learning analytics or marketing literature. The specific challenge of deploying lead scoring models in growth-stage EdTech companies with limited data infrastructure and LatAm-specific market characteristics represents a contextual contribution not covered by existing work.

From a methodological point of view, this thesis also contributes to the question of how binary classification models should be evaluated and deployed when the operational objective is not accuracy maximisation but ranking and maximal finding of attendance segment. The use of probability bucketing and threshold calibration to operational objectives, rather than default classification thresholds, are consistent with recommendations in the imbalanced learning literature [19], but has received limited attention in the specific context of EdTech acquisition funnels. The visualisation of model outputs through a BI dashboard, highlighting surfacing priority levels to sales and operations teams, further contributes to the arising literature on human-in-the-loop decision support in marketing operations [32]. This thesis therefore occupies a genuine research gap at the intersection of no-show prediction, EdTech digital transformation, learning analytics, and operational ML deployment.

Chapter 4

Background

4.1 Key Acquisition Metrics

Digital acquisition is commonly represented as a sequence of measurable actions. A user may see an advertisement, visit a landing page, submit contact information, book a trial class, attend it, and purchase a course afterwards. Each action can be described through a volume metric, a conversion rate, or a cost metric. However, a funnel metric is meaningful only when its numerator, denominator, observation window, and unit of analysis are explicitly defined. For example, "conversion rate" may refer to click-to-lead conversion, lead-to-booking conversion, booking-to-attendance conversion and others. These measures describe different stages and should not be used interchangeably.

Let I denote the number of advertising impressions, C the number of tracked clicks, L the number of generated leads, B the number of bookings of trial classes, A the number of attended trial classes, P the number of new paying customers, and S the marketing expenditure of the campaign for the observed period. These quantities allow to define the main funnel metrics consistently.

The click-through rate measures the initial step of the marketing funnel:

$$\mathrm{CTR} = \frac{C}{I}. \tag{4.1}$$

The click-to-lead conversion rate measures the proportion of users who submitted the lead form:

$$\mathrm{CR}_{\text{click-to-lead}} = \frac{L}{C}. \tag{4.2}$$

Cost per lead measures how much the company pays for one acquired lead:

$$\mathrm{CPL} = \frac{S}{L}. \tag{4.3}$$

Although CPL is an efficiency measure, it does not completely represent lead quality. A campaign may generate cheap leads while producing weak booking, attendance, or payment outcomes. Marketing metrics should therefore be interpreted as a connected system rather than as isolated optimisation targets [3].

The lead-to-booking conversion rate is defined as:

$$\mathrm{CR}_{\text{lead-to-booking}} = \frac{B}{L}. \tag{4.4}$$

Booking indicates that a lead is interested in a trial class, but it is not equivalent to actual participation. The metric directly aligned with this thesis is the booking-to-attendance conversion rate:

$$\mathrm{AR} = \mathrm{CR}_{\text{booking-to-attendance}} = \frac{A}{B}. \tag{4.5}$$

The scoring model estimates each booked lead's probability of attendance based on their observed characteristics, while the attendance rate represents the corresponding aggregate outcome. Upstream metrics such as CTR and CPL describe how traffic and leads are acquired, whereas attendance rate describes the intermediate outcome of the acquisition process.

Downstream conversion may be calculated from different funnel stages. The attendance-to-payment rate is:

$$\mathrm{CR}_{\text{attendance-to-payment}} = \frac{P}{A}, \tag{4.6}$$

And the end-to-end lead-to-payment rate is:

$$\mathrm{CR}_{\text{lead-to-payment}} = \frac{P}{L}. \tag{4.7}$$

The first measure is more closely associated with the effectiveness of the trial class and subsequent sales process. The second combines acquisition, booking, attendance, and sales conversion. A simplified customer acquisition cost is:

$$\mathrm{CAC} = \frac{S}{P}. \tag{4.8}$$

In managerial reporting, the numerator may include broader sales and marketing costs rather than media spend alone. The cost boundary must therefore be stated whenever CAC is compared across teams, markets, or time periods [3].

Described metrics represent three analytically distinct dimensions. Volume describes how many prospective customers enter a stage. Funnel efficiency describes the proportion of following stages. Lead quality describes the probability that a lead produces a valuable downstream result. Higher lead volume or lower CPL does not necessarily imply better business performance if attendance and payment probabilities decrease.

Lead scoring adds an individual-level estimate to these aggregate metrics. Funnel reporting may indicate that a campaign has a particular attendance rate, but it does not differentiate among booked leads within that group. A predictive model estimates each lead's probability of attendance conditional on their observed characteristics, enabling leads at the same funnel stage to be ranked or segmented according to their predicted likelihood of attending.

Let $y_i \in \{0, 1\}$ denote the observed attendance outcome for lead i, and let N be the number of eligible booked leads. The overall attendance rate is:

$$\pi = \frac{1}{N} \sum_{i=1}^{N} y_i. \tag{4.9}$$

For an bucketed score segment g containing N_g leads, its observed attendance rate is:

$$\mathrm{AR}_g = \frac{1}{N_g} \sum_{i \in g} y_i. \tag{4.10}$$

The segment lift relative to the complete result is:

$$\mathrm{Lift}_g = \frac{\mathrm{AR}_g}{\pi}. \tag{4.11}$$

A lift above one indicates that the segment contains a higher concentration of attenders than the unsegmented booked-lead population. This measure links statistical ranking to an operational funnel outcome. However, lift must be evaluated on unseen observations and reported together with segment size: a very small segment may achieve high lift while capturing only a limited share of all attending leads.

4.2 Machine Learning Foundations

The available data is structured and heterogeneous: it includes different formats and categories. Some effects may be approximately additive, while others may depend on thresholds or interactions. The modelling strategy therefore requires a simple benchmark and more flexible nonlinear alternatives.

A linear baseline establishes whether the available variables contain a stable signal that can be represented by an additive decision function. Tree-based ensembles test whether nonlinearities and interactions add out-of-sample predictive value. The comparison is methodological rather than hierarchical: a more complex algorithm is justified only if its improvement is sufficiently large and stable to compensate for additional tuning and interpretability interpretation.

4.2.1 Binary Classification Framework

In probabilistic binary classification, a model $f_{\boldsymbol{\theta}}$ maps a feature vector to an estimated probability:

$$\widehat{p}_i = f_{\boldsymbol{\theta}}(\mathbf{x}_i), \tag{4.12}$$

where $\boldsymbol{\theta}$ denotes the fitted parameters. Training estimates these parameters by minimising an empirical loss over a training sample. For probabilistic binary classifiers, a common objective is binary cross-entropy, or log loss:

$$\mathcal{L}_{\log} = -\frac{1}{N}\sum_{i=1}^{N}\left[y_i \log(\widehat{p}_i) + (1-y_i)\log(1-\widehat{p}_i)\right]. \tag{4.13}$$

Log loss penalises confident incorrect predictions more strongly than uncertain ones. It therefore evaluates the probability assigned to the realised class rather than only the correctness of a thresholded label [18].

Model development requires a separation between fitting and evaluation. Training observations are used to estimate model parameters. Validation observations support hyperparameter selection, preprocessing decisions, and threshold design. A final test set, or an equivalent temporally later holdout, is reserved for estimating generalisation after these choices have been made. Repeatedly choosing configurations based on the test set would make the test result part of the modelling process and bias the reported performance upward.

In this application, generalisation has a temporal and organisational meaning. The model is expected to score future booked leads, not to reproduce historical labels. Its performance can change when campaign composition, quiz design, tracking coverage, CRM logic, or user behaviour changes. An out-of-sample evaluation is therefore a minimum requirement rather than a guarantee of permanent validity.

Preprocessing depends on the model family. Linear models require numerical inputs and commonly use one-hot encoding for categorical variables. Numerical scaling can also improve optimisation and make regularisation more comparable across coefficients. Tree-based methods do not require variables to share a common scale, and some gradient-boosting implementations provide specialised handling of categorical values. These differences influence both pipeline design and the relationships that can be learned.

4.2.2 Linear Baselines and Limitations

Logistic regression is a natural baseline because it directly models a binary probability and offers a comparatively transparent parameterisation. It assumes that the log-odds of attendance are an additive linear function of the predictors:

$$\log\left(\frac{p_i}{1-p_i}\right) = \beta_0 + \sum_{j=1}^{m} \beta_j x_{ij}. \tag{4.14}$$

The probability is obtained through the logistic function:

$$\widehat{p_i} = \frac{1}{1 + \exp\left[-\left(\beta_0 + \sum_{j=1}^{m} \beta_j x_{ij}\right)\right]}. \tag{4.15}$$

Holding other variables constant, a one-unit increase in x_j changes the log-odds by β_j and multiplies the odds by $\exp(\beta_j)$. This interpretation must nevertheless be treated cautiously when variables are transformed, highly correlated, or represented by sets of dummy indicators [18].

Regularisation constrains coefficient magnitude and can improve stability when the feature space is large relative to the sample size. With an L_2 penalty, the training objective becomes

$$\mathcal{L}_{L_2} = \mathcal{L}_{\log} + \lambda \sum_{j=1}^{m} \beta_j^2, \tag{4.16}$$

whereas an L_1 penalty gives

$$\mathcal{L}_{L_1} = \mathcal{L}_{\log} + \lambda \sum_{j=1}^{m} |\beta_j|. \tag{4.17}$$

The hyperparameter λ controls the trade-off between fitting the training observations and shrinking the coefficient vector. L_1 regularisation can set some coefficients to zero, while L_2 regularisation typically shrinks them continuously [18].

The principal limitation of the baseline is its functional form. A standard logistic model assumes an additive linear relationship between each numerical predictor and the log-odds. If attendance probability changes nonlinearly with booking time, time of day, or response duration, the relevant transformations must be introduced manually. Likewise, interactions must be specified explicitly. For example, the relationship between a quiz response and attendance may differ by acquisition source or market, but a main-effects-only specification cannot discover this conditional structure.

Categorical variables create an additional practical constraint. One-hot encoding can produce a wide and sparse matrix, especially when categories are numerous. Rare levels may lead to unstable estimates, while unseen levels require an explicit handling rule. Missing values also need to be processed before fitting. These limitations do not make logistic regression unsuitable; rather, they define the conditions under which a nonlinear model may provide additional value.

The linear model therefore serves as a substantive benchmark. If a tree ensemble produces only a negligible improvement on unseen data, the simpler specification may be preferable. A material and stable improvement, by contrast, suggests that nonlinear effects, interactions, or alternative categorical handling contain information that the baseline does not capture.

4.2.3 Decision Trees and Tree-Based Ensembles

A decision tree is a supervised classification technique that allows to represent a set of classification rules with a tree. It recursively partitions the feature space into increasingly homogeneous regions. Starting from the root node, the algorithm evaluates candidate split rules and selects the rule that produces the greatest improvement according to a predefined criterion. For a numerical variable, a split may take the form $x_j \leq c$, where c is a threshold. For a categorical variable, the rule separates observations using n-ary or binary split.

In binary classification, the quality of a candidate split is commonly assessed using an impurity measure, such as Gini impurity or entropy. Gini impurity for a node t formula:

$$G(t) = 1 - \sum_{k=1}^{K} p_{k,t}^2,$$

where $p_{k,t}$ is the proportion of observations from class k in node t. For the binary case, $K = 2$.

A node containing observations from only one class has an impurity of zero, whereas a node containing an equal proportion of the two classes has the highest impurity. The selected split is the one that produces the largest weighted reduction in impurity between the parent node and its child nodes.

The partitioning process continues until a stopping condition is reached. Such conditions may include a maximum tree depth, a minimum number of observations required to split a node, a minimum number of observations in a leaf, or the absence of sufficient improvement of purity measure. Each terminal node, or leaf, is assigned a prediction based on the training observations that reach it. In probabilistic binary classification, the prediction of a leaf is typically derived from the proportion of positive observations within that leaf:

$$\hat{p}_t = \frac{N_{t,1}}{N_t},$$

where $N_{t,1}$ is the number of positive observations in leaf t, and N_t is the total number of observations assigned to that leaf.

Decision trees are able to represent nonlinear relationships, threshold effects, and interactions without requiring these structures to be specified in advance. For example, the relationship between booking time and attendance may not be linear. Attendance

probability may change only after a particular time threshold. Similarly, a timing variable may affect the prediction only for leads originating from a specific acquisition source. A tree can represent this interaction through sequential splits: the first split may separate leads by source, while a later split applies the timing condition only within one of the resulting branches.

This flexibility makes decision trees suitable for heterogeneous tabular data. They do not require numerical variables to be standardised, and their output is easily interpreted by users. However, an individual tree also has significant limitations. A sufficiently deep tree can create highly specific areas containing only a small number of observations. It may therefore capture noise or sample-specific patterns rather than relationships that generalise to new data.

Individual decision trees can also have high variance. Relatively small changes in the training set may alter the selected root split and consequently change much of the subsequent tree structure. Restricting tree depth, increasing the minimum leaf size, or pruning the tree can reduce this instability, but these constraints may also reduce the model's ability to capture relevant patterns.

Tree-based ensemble methods address this limitation by combining multiple decision trees. The main ensemble approaches differ in how the trees are trained and how observations are treated during training. In bagging-based methods, trees are fitted independently on different samples of the training data and their predictions are aggregated. The observations do not receive adaptive importance weights based on the errors of previous trees. And vice versa, in boosting methods, trees are fitted sequentially, and each new tree is constructed to improve the errors of the previous trees.

Some boosting algorithms explicitly change observation weights. AdaBoost, for example, increases the weights of incorrectly classified observations so that later models focus more strongly on them. Gradient boosting follows a related but technically different principle. Instead of directly assigning larger weights to all misclassified observations, each new tree is fitted to the negative gradient of the loss function, which represents the direction in which the current predictions should be corrected. Thus, both approaches focus successive learners on previous errors, but they do so through different optimisation mechanisms.

Random Forest is a bagging-based tree ensemble. Each tree is trained on a bootstrap sample drawn from the original training data with replacement. As a result, some observations may appear multiple times in the sample used for one tree, while others may not appear in that tree's training sample.

Random Forest also introduces randomness at the feature level. At each split, the algorithm evaluates only a randomly selected subset of the available predictors rather than considering all features. This reduces the probability that the same dominant variable determines the upper structure of every tree. The resulting trees are therefore less correlated with one another than ordinary bagged trees.

For probabilistic classification, the Random Forest prediction can be expressed as the average of the probabilities generated by its M individual trees:

$$\hat{p}(\mathbf{x}) = \frac{1}{M} \sum_{m=1}^{M} \hat{p}_m(\mathbf{x}),$$

where M is the number of trees. Randomisation reduces correlation among the component trees and can lower variance relative to an individual tree [7].

Main Random Forest hyperparameters include the number of trees, maximum tree depth, number of candidate features considered at each split, minimum number of observations required to split a node, and minimum leaf size. Increasing the number of trees generally stabilises the ensemble prediction, although it also increases computational cost. Tree complexity parameters control the balance between capturing detailed patterns and avoiding overfitting.

Gradient boosting constructs an ensemble sequentially. The model begins with an initial prediction and adds trees one at a time:

$$F_M(\mathbf{x}) = F_0(\mathbf{x}) + \sum_{m=1}^{M} \eta h_m(\mathbf{x}),$$

where $h_m(\mathbf{x})$ is the tree added at iteration m, and η is the learning rate. Each new tree is fitted to correct the current ensemble according to the selected loss function. In binary classification, the optimisation is commonly based on log loss.

The learning rate determines the contribution of each newly added tree. A smaller learning rate usually requires a larger number of trees but may produce more gradual and stable learning. Other important parameters include the number of boosting iterations, tree depth, minimum leaf size, row and feature subsampling, and regularisation strength.

Modern implementations of gradient-boosted trees differ mainly in optimisation strategy, computational efficiency, regularisation, and categorical-feature handling.

LightGBM uses histogram-based split finding together with Gradient-based One-Side Sampling and Exclusive Feature Bundling to reduce computational cost in high-dimensional or large datasets [23]. These methods are implementations of the gradient-boosting framework rather than fundamentally different prediction tasks.

CatBoost is designed for datasets containing categorical variables. It uses ordered target statistics and ordered boosting procedures to reduce prediction shift and target leakage that can arise when categorical encodings are constructed using the target from the same observation [37]. This design is relevant when the feature set contains market, device, source, or quiz-response categories. It does not remove the need for a correct global prediction cutoff: algorithmic protection against leakage in categorical encoding cannot compensate for business variables recorded after the intended scoring time.

Tree-based ensembles are suitable candidates for the attendance problem because they can combine heterogeneous predictors and represent nonlinear conditional relationships. Their flexibility also creates two requirements. First, complexity must be controlled and assessed on unseen data. Second, the fitted model is less directly interpretable than a sparse linear equation. Split-based importance alone does not show how a value affected a specific lead's prediction and may be biased toward variables with many possible split points.

A final distinction concerns discrimination and calibration. A boosted model may rank attenders above non-attenders effectively while producing probabilities that are systematically too high or too low. Probability calibration must therefore be evaluated separately whenever score magnitudes or probability buckets are used operationally [35].

4.2.4 Evaluation Metrics

To fully describe the quality of a probabilistic lead-scoring model, several categories of metrics need to be calculated. Threshold-based metrics evaluate a selected decision rule, ranking metrics assess the ordering of observations, probability metrics assess the numerical quality of $\widehat{p}_i$, and business metrics assess whether the ranking produces useful operational segments.

At a chosen threshold, predictions form a confusion matrix. True positives (TP) are attending leads correctly classified as positive; false positives (FP) are non-attenders classified as positive; true negatives (TN) are non-attenders correctly classified as negative; and false negatives (FN) are attenders classified as negative.

Accuracy is

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}. \tag{4.18}$$

When the positive class is less frequent, accuracy can be dominated by correct negative predictions. A majority-class model may consequently appear accurate while identifying no attenders. Accuracy is therefore reported only as a supplementary measure in this setting.

Precision is

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}. \tag{4.19}$$

For an operational priority group, precision answers the question: among leads selected for higher-priority treatment, what proportion attended? Recall is

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}. \tag{4.20}$$

Recall answers a different question: what proportion of all attending leads was captured by the selected group? Expanding the priority group commonly increases

recall but may reduce precision. The appropriate balance depends on follow-up capacity and the relative costs.

For the lead-attendance prediction task, recall has a particular operational importance because the objective is to identify as many actual attenders as possible within the prioritised population. A false negative represents an attending lead who was not identified as high priority and might therefore receive a lower level of follow-up treatment. By contrast, the direct cost of a false positive is comparatively limited: a non-attending lead may receive an unnecessary confirmation call or reminder message. Since the operational cost of an additional call or SMS is lower than the cost of failing to prioritise a lead who is likely to attend, model and threshold selection should place relatively greater emphasis on recall. Nevertheless, precision remains relevant because very low precision would direct a large share of the available follow-up capacity towards non-attending leads.

The F_1 score is the harmonic mean of precision and recall:

$$F_1 = 2\frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{4.21}$$

Although convenient, F_1 assigns equal importance to precision and recall and remains threshold-dependent. This equal weighting does not fully reflect the operational priorities of the present task, in which recall is considered more important than precision. The F_1 score should therefore not replace the separate reporting of its two components.

The receiver operating characteristic curve plots the true-positive rate against the false-positive rate across all thresholds. ROC-AUC summarises discrimination: it can be interpreted as the probability that a randomly selected positive observation receives a higher score than a randomly selected negative observation, subject to the usual treatment of ties [16]. A value of 0.5 corresponds to random ordering, while a value of 1 represents perfect separation.

ROC-AUC is threshold-independent, but it may provide an incomplete view in imbalanced settings because the false-positive rate is normalised by the total number of negatives. The precision–recall curve focuses directly on positive-class retrieval and the purity of the selected positive set. Its baseline precision equals positive-class prevalence, making it sensitive to the class distribution. Precision–recall analysis is therefore particularly informative for minority-outcome classification [41].

Ranking quality does not ensure probability quality. Calibration describes whether predicted probabilities correspond to observed frequencies. A well-calibrated model should produce an attendance rate close to q among observations assigned probabilities around q. Reliability diagrams compare mean predicted probability with observed outcome frequency across probability intervals. They should be interpreted alongside discrimination rather than as a substitute for it [35].

The Brier score evaluates squared probability error:

$$\text{Brier} = \frac{1}{N}\sum_{i=1}^{N}(\widehat{p}_i - y_i)^2. \tag{4.22}$$

Lower values indicate smaller average squared differences between forecasts and binary outcomes [8]. Log loss is another probability-sensitive metric, with a stronger penalty for highly confident errors. Neither score should be described as a pure calibration measure, because both reflect multiple aspects of probabilistic performance.

Operational evaluation should additionally reflect the team's finite capacity. If leads are ordered from highest to lowest predicted probability, precision at the top k per cent is

$$\text{Precision@}k = \frac{\text{attenders among the top } k\% \text{ of scores}}{\text{number of leads in the top } k\%}, \tag{4.23}$$

and recall at the top k per cent is

$$\text{Recall@}k = \frac{\text{attenders among the top } k\% \text{ of scores}}{\text{all attenders}}. \tag{4.24}$$

These measures answer whether the model concentrates attenders within the proportion of leads that can realistically receive enhanced follow-up. Segment attendance and lift provide the corresponding evaluation for low-, medium-, and high-score groups. A useful ordering should produce a monotonic increase in conversion rates:

$$\text{AR}_{\text{Low}} < \text{AR}_{\text{Medium}} < \text{AR}_{\text{High}}, \tag{4.25}$$

on data not used to select the thresholds. Segment size, confidence intervals, and temporal stability should accompany these rates in the empirical results.

4.3 Interpretability: SHAP Framework

Interpretability is required for both analytical validation and operational communication. Logistic-regression coefficients provide a direct description of an additive model on the log-odds scale, but a tree ensemble combines many conditional rules and interactions. Its predictions cannot be adequately explained by inspecting a single tree or by reporting only split-based feature importance.

SHAP, or SHapley Additive exPlanations, is a feature-attribution framework based on Shapley values from cooperative game theory. It represents a prediction as a baseline plus additive feature contributions [31]. For observation i,

$$f(\mathbf{x}_i) = \phi_0 + \sum_{j=1}^{m}\phi_{ij}, \tag{4.26}$$

where ϕ_0 is the expected model output under the explainer's background distribution and ϕ_{ij} is the contribution assigned to feature j for observation i. A positive value moves the explained model output above the baseline, while a negative value moves it below the baseline.

The output scale must be identified explicitly. Depending on the model and explainer configuration, binary-classification SHAP values may decompose raw margin or log-odds rather than probability. A SHAP value of 0.2 on the log-odds scale is not a direct increase of 20 percentage points in attendance probability. Probability-scale statements should be made only when the explainer has been configured and validated to produce that scale.

SHAP supports local and global analysis. A local explanation decomposes the score for one lead, identifying the features that moved the model output relative to the baseline. Such an explanation can help an analyst inspect whether a particular score was driven by legitimate pre-prediction information or by a suspicious variable.

Global summaries aggregate local attributions. A common importance measure is the mean absolute SHAP value:

$$I_j^{\mathrm{SHAP}} = \frac{1}{N} \sum_{i=1}^{N} |\phi_{ij}|. \tag{4.27}$$

It measures average attribution magnitude, not direction. A feature may have high importance because different values strongly increase predictions for some observations and decrease them for others. A SHAP summary plot preserves the sign and distribution of local contributions, while a dependence plot relates a feature's observed values to its attributed contribution and can reveal thresholds or heterogeneous effects.

For tree ensembles, TreeSHAP provides algorithms for computing game-theoretic attributions efficiently and for combining local explanations into global descriptions while retaining consistency with the underlying predictions [28]. This makes the framework suitable for inspecting Random Forest and gradient-boosted models used on structured acquisition data.

In this thesis, SHAP has four roles. First, it provides a plausibility audit: highly influential features are checked against business definitions and data timing. A variable that should not exist at the scoring point may reveal leakage; an identifier with high attribution may indicate memorisation or a data artefact. Second, signed contributions show whether observed values move the model output upward or downward rather than merely indicating that a feature was used. Third, distributions of explanations can be compared across score buckets or operational subgroups to identify heterogeneous model behaviour. Fourth, local explanations support communication of individual predictions to analysts without reducing the model to a single global importance ranking.

These benefits do not make SHAP a causal method. A positive attribution states that a feature increased the fitted model's output relative to the selected background

expectation. It does not establish that intervening on the feature would increase attendance. The attribution is conditional on the model, the training data, and the explainer's assumptions.

Feature dependence is a further limitation. When predictors are correlated, the allocation of credit among them depends on how missing-feature coalitions and the background distribution are represented. Methods that assume feature independence can evaluate implausible combinations and produce misleading explanations. Dependence-aware extensions have therefore been proposed for more accurate Shapley-value approximation when features are correlated [1].

Finally, SHAP explains the fitted model rather than proving that the model is correct. A faithful explanation can still describe a prediction learned from biased, unstable, or incorrectly constructed data. Attribution analysis must therefore be combined with out-of-sample evaluation, leakage prevention, subgroup checks, and monitoring of input and outcome distributions.

Within this study, SHAP is consequently used as an analytical audit and communication framework. It helps identify how the selected model uses admissible features and whether the resulting patterns are substantively plausible. It does not replace predictive validation, operational judgement, or causal analysis.

Chapter 5

Data Analysis

5.1 Data Acquisition Pipeline

The dataset for analysis was constructed by integrating information generated at different stages of the Kodland acquisition and trial-class booking process.

The objective of the data acquisition pipeline was to create a modelling table containing the information available for each eligible quiz completion before the attendance outcome became known. The resulting dataset combined qualification quiz responses, CRM and booking records, historical Google Analytics signals, aggregated client-level information, and marketing campaign metadata.

The observation period covered quiz completions recorded between 27 October 2025 and 12 March 2026. This period should be distinguished from the model development period. The underlying customer and campaign events occurred during the observation window, whereas the analytical dataset was assembled and the predictive model was developed in March 2026 as part of the internship project. Thus, March 2026 represents the time of retrospective data extraction, analysis, and model construction rather than the period in which the modelled events occurred.

SQL in Amazon Redshift was used to select the eligible population, join the source tables, construct the target variable, and calculate several derived features. The resulting analytical table was then exported and loaded into Python for data-quality analysis, preprocessing, feature engineering, model training, and evaluation. Constructing the main modelling table in the data warehouse ensured that the same eligibility criteria, join conditions, and temporal restrictions were applied consistently across observations.

5.1.1 Analytical Unit and Population Description

The analytical unit was defined as a combination of a lead and an individual quiz completion. Consequently, the same CRM lead could appear in more than one row if the

lead completed the qualification quiz multiple times. Each quiz submission was retained as a separate observation rather than reducing the data to one row per CRM identifier.

This distinction explains why the number of rows in the dataset was slightly larger than the number of distinct CRM identifiers. The final exported dataset contained 5,690 observations and 5,681 distinct values of `amocrm_id`. The repeated identifiers therefore did not necessarily represent accidental duplicates. Instead, they could correspond to separate quiz completions associated with the same lead.

The modelling population was restricted according to four principal eligibility criteria. Firstly, the users who saw the advertising banner were located in Latin America. Secondly, the lead should have completed the main qualification quiz. Thirdly, the quiz contained exactly 12 recorded questions, corresponding to the complete expected quiz structure. Lastly, the quiz completion could be linked to a known CRM user identifier.

Restricting the dataset to quiz records with exactly 12 questions excluded incomplete submissions and records created under alternative or technically inconsistent quiz structures. This improved comparability between observations because each included respondent had been exposed to the same expected qualification flow.

The dataset contained 62 columns at the point of export. All timestamps were stored and processed in Coordinated Universal Time (UTC), reflecting the time at which events entered the central database. Using a common timezone avoided inconsistencies caused by combining observations from multiple Latin American markets.

5.1.2 Source Systems

The final table was constructed from five principal source groups: the quiz table, CRM table, Google Analytics table, campaign metadata table, and aggregated client table. Each source described a different part of the acquisition and trial-class process. In figure 5.1, all table sources are described.

The quiz table formed the starting point of the pipeline because the analytical unit was centred on an individual quiz completion. It provided the quiz timestamps, the number of submitted questions, and qualification responses related to factors such as decision-making responsibility, expected budget, access to a computer, and internet stability.

The quiz records were joined to the CRM table using `booking_id`. The CRM table contained information about the booked trial class and its subsequent status. It also included the complete history of clients' booking statuses, allowing changes in status over time to be tracked.

Google Analytics data were connected to the quiz records through `client_id`. This source provided historical digital interaction and device information, including previous sessions, operating system, device type, device brand, traffic source, and UTM-related attributes.

Source	Join key	Information provided
Quiz table	booking_id and client_id	Quiz start and completion timestamps, number of questions, qualification answers, and the identifiers required to connect the quiz to CRM and web data.
CRM table	booking_id	Booking records, trial-class timeslots, rescheduling information, booking status, course information, and the attendance outcome.
Google Analytics table	client_id	Historical web interactions, session information, device characteristics, and other digital signals collected before quiz completion.
Campaign metadata table	ad_id	Campaign and banner parameters, creative characteristics together with aggregated advertising performance measures.
Aggregated client table	client_id	Consolidated client-level CRM characteristics that could be associated with the quiz and booking observation.

Figure 5.1: Data sources used in the acquisition pipeline

The campaign metadata table was connected through ad_id. It provided information about the campaign, advertisement, creative format, offer, product, bidding strategy, and aggregated performance measures such as impressions, link clicks, and advertising spend.

Finally, the aggregated client table was joined through crm_client_id. This table consolidated information recorded at the client level and allowed the pipeline to attach relevant CRM characteristics without repeatedly processing multiple operational records for the same client.

5.1.3 Temporal Filtering

A central requirement of the pipeline was temporal validity. Only information available by the time of quiz completion could be used to construct predictive variables. This condition was particularly important for Google Analytics and campaign data because both sources continued to accumulate information after the lead had completed the quiz.

For the Google Analytics source, only sessions and events recorded before quiz completion were eligible for inclusion. Variables such as whether the user had previous sessions and the number of days since the last observed session were therefore calculated relative to the quiz completion timestamp.

Campaign-level metrics were also restricted by date. Impressions, link clicks, and advertising spend were aggregated only from campaign records available before the corresponding quiz completion. This prevented the model from using future campaign

performance that would not have been known when the lead entered the scoring process.

The campaign metrics represented aggregated campaign-level context rather than individual user behaviour. For example, the value of `impressions` attached to a row did not indicate how many times the particular lead had viewed an advertisement. It described the cumulative or aggregated performance of the associated campaign before the quiz completion. Multiple leads associated with the same advertisement could therefore have similar or identical campaign-level values.

This temporal filtering was necessary to prevent data leakage. Without it, a lead completing the quiz at an earlier date could receive campaign statistics accumulated after their quiz submission. The model would then use information from the future, leading to an unrealistic estimate of out-of-sample performance.

5.2 Feature Description

The final feature matrix was a combination of raw variables from sources described in the previous sections and engineered values. Below is a detailed description of each variable:

5.2.1 CRM, Booking, and Scheduling Features

CRM and booking variables describe the operational state of the trial-class appointment. They were obtained from the CRM table and from booking-related fields linked to the quiz record through `booking_id`.

- `amocrm_id` is a pseudonymised CRM identifier. It was used for record linkage, duplicate checks, and validation of repeated quiz completions. It was not treated as a behavioural feature for modelling because it identifies an entity rather than describing a generalisable lead characteristic.
- `first_booking_timeslot` records the first trial-class timeslot associated with the quiz completion. It represents the initial appointment selected by the lead or assigned through the booking process.
- `last_booking_timeslot` records the most recent trial-class timeslot available in the CRM record. It is equal to `first_booking_timeslot` when no rescheduling occurred and differs from it when the trial class was moved to another time.
- `last_booking_status` is the latest CRM status of the booking. It is useful for target validation and descriptive analysis, but it must be handled carefully in predictive modelling because final status information may be observed only after the scoring moment.

- stud_age_amo is the student age recorded in the CRM system. It represents a CRM-side demographic characteristic and may differ from quiz-based age fields if the information was entered or transformed through a different process.
- marketing_course_global describes the course in the marketing layer. It indicates which educational offer was promoted or assigned to the lead.
- course_global describes the course category in the integrated dataset. It can be used to compare attendance behaviour across product lines such as coding, media, or mathematics-oriented offers.

Several timing variables were derived from CRM and quiz timestamps:

- hour_first_booking_timeslot extracts the hour of the first booked trial-class timeslot. It represents the time of day at which the class was scheduled.
- gap_lead_to_mk_hours measures the time interval, in hours, between lead or quiz creation and the scheduled trial-class timeslot. It captures how far into the future the appointment was set.
- days_to_mk is the same scheduling distance expressed in days. It is easier to interpret operationally because trial-class attendance often depends on whether the class is scheduled for the same day, the next day, or several days later.
- day_of_week_tc identifies the weekday of the scheduled trial class. It captures possible differences between weekday and weekend attendance behaviour.

These variables are directly related to the booking process and are theoretically relevant because no-show behaviour is often affected by the time gap between commitment and attendance. However, only those booking variables that would be available at the scoring moment should be used as model inputs.

5.2.2 Qualification Quiz and Session Behaviour Features

Quiz variables were obtained from the qualification quiz table. They describe either the user's interaction with the quiz interface or the answers provided during the qualification flow.

- time_started is the timestamp at which the quiz was started.
- time_end is the timestamp at which the quiz was completed.
- quiz_completion_time_seconds is the duration of the quiz completion process. It is derived from time_started and time_end. Very short or very long durations may indicate different levels of engagement, hesitation or distraction.

- `questions_count` records the number of quiz questions or logged quiz steps associated with the observation. It was used as a consistency check for the quiz structure. Because the recorded values are not fully constant in the exported dataset, this field should be interpreted as a technical quality-control variable rather than as a substantive behavioural predictor without further validation.
- `day_of_week_quiz_start` is derived from `time_started`. It indicates the weekday on which the quiz interaction began and may capture differences in user behaviour across the week.

The quiz answers describe the student, the parent or respondent, and the family's readiness for online classes:

- `respondent` identifies who completed the quiz or represented the decision-making side, such as a parent or another respondent category.
- `age` records the age value submitted through the quiz flow. It is conceptually related to the student's age but originates from the quiz rather than from the CRM system.
- `gender` records the gender category provided in the quiz.
- `decision_responsibility` describes who is responsible for the decision about enrolling in the course. This variable is relevant because attendance may depend not only on the student's interest but also on parental or household decision authority.
- `free_time` captures how child usually spends their free time: plays video-games, reads etc.
- `took_online_classes_before` indicates previous experience with online education. Prior experience may reduce uncertainty about the format and make trial attendance more likely.
- `reaction_to_trial_class` captures the respondent's stated reaction to the proposed trial class. It is an intent-related variable because it reflects the perceived attractiveness or acceptability of attending a free introductory session.
- `monthly_budget` captures the approximate monthly budget reported by the family for paid educational activities.
- `extracurricular_spend` records previous spending on extracurricular activities. This variable approximates prior willingness or ability to pay for non-school educational services.

- would_pay_if_useful measures whether the respondent would consider payment if the course were perceived as useful. It is a direct stated-intent variable related to commercial readiness.
- has_computer indicates whether the student has access to a computer or laptop, or only to a phone or tablet. This is important in an online-class context because some course formats may require a suitable device.
- internet_stability describes the respondent's assessment of connection stability. It reflects technical readiness for attending an online class.
- speed_internet is a numerical technical-readiness signal related to internet quality. It complements the categorical internet_stability variable and is measured by the means of Kodlands' IT system.

These quiz-derived variables are important because they capture information not available in standard advertising logs. They provide explicit signals about motivation, budget, household decision-making, and technical readiness.

5.2.3 Google Analytics, Device, and Acquisition Features

Google Analytics variables were attached to quiz records through client_id. They describe observed web behaviour, device characteristics, and attribution information available before quiz completion.

- had_sessions_before_quiz indicates whether Google Analytics recorded at least one previous session for the same client identifier before quiz completion. It captures whether the user had prior observable engagement with Kodland before submitting the quiz.
- days_since_last_ga_session_asof measures the time elapsed between the most recent observed Google Analytics session and quiz completion. A shorter interval may indicate more recent engagement, while missing values can indicate that no previous session was observed or could be linked.
- os_device describes the operating system or device environment detected by the tracking system.
- device_type classifies the user's device into broader technical categories, such as desktop or mobile device types.
- brand_device records the detected device brand where available.

- device_type_ui is a user-interface or tracking-side device classification. It may overlap with device_type but can originate from a different mapping or interface-level categorisation.

Geographic and acquisition variables describe where the lead came from and how the acquisition path was attributed:

- country identifies the country associated with the lead or acquisition event.
- region records a broader regional grouping or market-level categorisation.
- source describes the high-level acquisition source, such as Google Search, Facebook, organic traffic, or other source groups.
- utm_source, utm_medium, utm_campaign, utm_content, utm_referrer, and utm_marketing are UTM and attribution fields captured from marketing links or tracking parameters. They describe the source, medium, campaign, content, referrer, and marketing attribution path through which the lead entered the funnel.

The attribution variables are potentially informative but require careful preprocessing. Some of them have high cardinality, particularly utm_referrer, and may contain many rare values. Such fields can create sparse encodings and may not generalise well unless grouped, transformed, or excluded.

5.2.4 Marketing Campaign Metadata Features

Campaign metadata was joined through ad_id. These variables describe the advertising context associated with the lead. They do not represent individual user actions. Instead, they provide campaign-level information available before the quiz completion.

- impressions is the number of advertising impressions accumulated for the associated campaign or advertisement before the quiz completion cutoff.
- link_clicks is the number of recorded link clicks for the associated campaign or advertisement before the quiz completion cutoff.
- spend_usd is the advertising expenditure, expressed in US dollars, accumulated before the quiz completion cutoff.
- campaign_event_empiric describes the optimisation event used in the advertising setup.
- campaign_bid_strategy_empiric describes the bidding strategy associated with the campaign.

- `campaign_offer_empiric` describes the offer communicated at the campaign level.
- `product_offer_empiric` records the product offer associated with the advertisement or landing flow.
- `ad_product_empiric` describes the advertised product category.
- `ad_group_product` provides a product grouping at the advertisement-group level.
- `ad_back_empiric` describes background or back-side creative attributes where such metadata were available.
- `ad_hero_dynamic_empiric` describes the main hero element or dynamic creative component used in the advertisement.
- `creative_call_to_action_type` describes the type of call-to-action used in the creative.
- `creative_concept` describes the broader creative idea or concept.
- `creative_uvp` describes the unique value proposition communicated by the creative.
- `landing_id` identifies the landing page or landing-flow variant associated with the advertisement.
- `format` describes the advertising creative format, such as banner or video-based formats.

Because campaign metrics were aggregated only up to the quiz completion date, they were designed to avoid future-information leakage. However, many campaign metadata fields were missing for a large share of observations. This suggests that metadata coverage varied across campaigns, platforms, or historical periods. These variables therefore required specific missing-value treatment before modelling.

5.3 Exploratory Data Analysis

Exploratory data analysis was conducted to understand the structure of the dataset, find patterns in a data, evaluate the target distribution, identify missingness patterns, and evaluate overall quality of the dataset for future analysis.

5.3.1 Dataset Size and Target Distribution

The dataset contained 5,690 lead–quiz-completion observations and 61 substantive columns. No fully duplicated rows were found. However, the number of distinct CRM identifiers was slightly lower than the number of rows, which is consistent with the analytical unit being a quiz completion associated with a lead rather than a unique lead.

The target variable was moderately imbalanced. There were 4,377 observations in the non-visited class and 1,313 observations in the visited class. Thus, 76.92% of observations had `visited_tc_7d_target = 0`, while 23.08% had `visited_tc_7d_target = 1`. This imbalance does not make the dataset unusable, but it means that accuracy alone would be an insufficient evaluation metric. A model could achieve relatively high accuracy by favouring the majority class while failing to identify attending leads. The class distribution therefore also influenced the modelling strategy: candidate algorithms, hyperparameters, and decision thresholds had to be evaluated with respect to their ability to recover the minority positive class rather than only to maximise aggregate predictive accuracy.

5.3.2 Geographic, Traffic Source, and Customer Readiness Signals

The descriptive analysis indicated that the dataset was concentrated in several Latin American markets. Peru accounted for the largest share of observations, representing 31.16% of the dataset, followed by Colombia with 20.93%, Argentina with 13.01%, Ecuador with 11.37%, Chile with 8.77%, and Mexico with 6.94%. The remaining observations were distributed across smaller countries and residual categories.

Trial-class attendance within seven days varied across the main countries. Chile had the highest visited trial-class rate at 31.06%, followed by Peru at 26.51%, Ecuador at 24.57%, and Colombia at 24.35%. Argentina had a substantially lower visited rate of 11.89%. These differences suggest that market context may be relevant for attendance prediction. However, they should be interpreted descriptively rather than causally, since country may be correlated with campaign mix, traffic source, operational process, and course offering.

Traffic source distribution was strongly concentrated in Google Performance Max, represented in the dataset as `google_pmax`. This source accounted for 80.37% of observations and had a visited trial-class rate of 23.42%. Facebook represented 12.02% of observations and had a visited rate of 20.61%. Some smaller traffic sources showed higher visited rates, but their sample sizes were limited, making these estimates less stable. For modelling purposes, rare traffic sources were grouped and labeled as ”Unknown” to reduce noise and avoid overfitting to small categories. Rare categories were defined as those with fewer than 50 observations.

Several quiz-derived readiness and intent variables also showed meaningful descrip-

tive associations with attendance. Leads reporting stable internet had a visited rate of 25.41%, compared with 13.35% for leads whose internet sometimes interrupted and 8.96% for leads reporting frequent interruptions. Device availability followed a similar pattern: leads with access to a computer or laptop had a visited rate of 26.42%, whereas leads using only a phone or tablet had a visited rate of 10.27%. Since trial classes are conducted online and may require active participation, these differences suggest that technical readiness is an important candidate predictor.

Commercial-intent variables showed additional variation. Leads whose families were already considering payment had a visited rate of 26.72%. Leads whose families would pay only if the price was reasonable had a visited rate of 23.36%, while those needing more information had a rate of 20.32%. In contrast, leads whose families stated that they did not plan to pay even if they liked the course had a visited rate of only 9.35%. This pattern suggests that stated willingness to pay is associated not only with downstream purchase potential, but also with short-term trial-class attendance.

Budget-related variables provided a similar signal. The largest monthly budget group, "up to 30 USD", represented 73.08% of observations and had a visited rate of 22.61%. The 30–60 USD group had a higher visited rate of 28.28%. For previous extracurricular spending, the 50–100 USD and 20–50 USD groups had visited rates of 28.39% and 25.95%, respectively, compared with 20.74% for families that previously reported no extracurricular spending.

Figure 5.2 presents selected categorical features by category frequency and visited trial-class rate. The bars show the total number of observations in each category, while the line indicates the share of observations with `visited_tc_7d_target = 1`. This representation makes it possible to compare both the size of each category and its observed association with short-term trial-class attendance.

Overall, geographic context, traffic source, technical readiness, and stated commercial intent all showed descriptive associations with short-term trial-class attendance. These findings support the inclusion of market, acquisition, and customer-profile variables in the modelling process. However, this analysis of differences is descriptive only. No statistical test was conducted to assess whether the observed differences between groups were statistically significant, such as a chi-squared test. Furthermore, the observed differences should not be interpreted as causal effects. They indicate associations in the available data rather than evidence that changing a category value would directly change attendance probability.

5.3.3 Data Quality Analysis

Data quality analysis focused on three types of issues that could affect subsequent preprocessing and model development: missing values, temporal consistency of timestamp-based variables, and extreme values in numerical features. These checks were necessary because the dataset was constructed from several source systems, including the quiz ta-

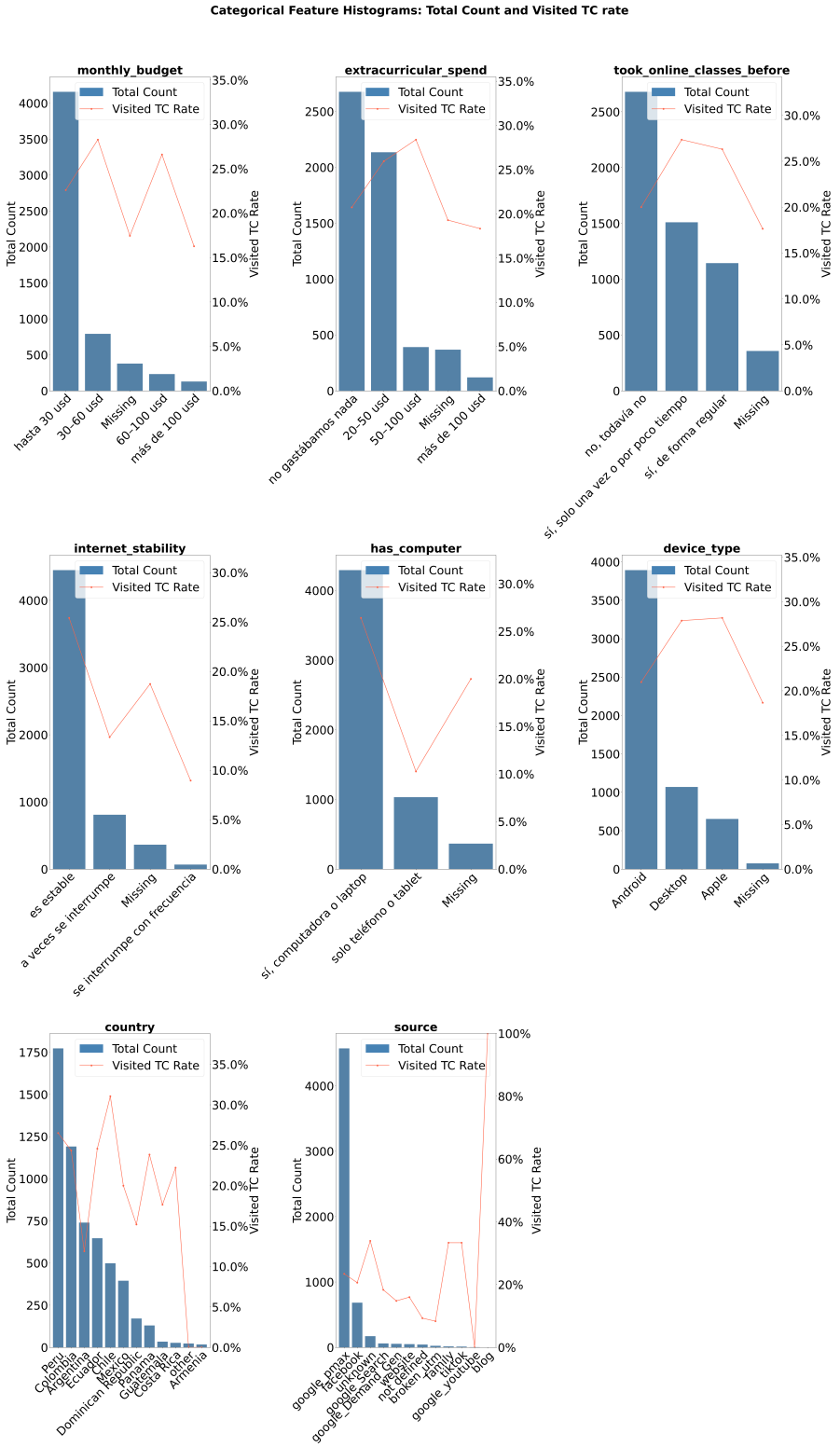

Figure 5.2: Distribution of selected categorical features and visited trial-class rate

ble, CRM table, Google Analytics table, campaign metadata table, and aggregated client table. As a result, some inconsistencies could arise from incomplete source coverage, tracking limitations, timestamp alignment, or differences in the level of aggregation.

Missing values were unevenly distributed across feature groups. The highest levels of missingness were observed among advertising creative metadata fields. Figure 5.3 shows the variables with the largest share of missing values.

creative_call_to_action_type	99.93
ad_back_empiric	96.17
ad_hero_dynamic_empiric	95.54
creative_concept	88.26
ad_group_product	87.70
days_since_last_ga_session_asof	87.45
creative_uvp	87.05
ad_product_empiric	86.98
landing_id	86.98
format	86.98
utm_content	84.34

Figure 5.3: Features with the highest missing-value percentages

The high missingness among creative-level campaign fields suggests that many advertising metadata attributes were available only for a subset of advertisements. This pattern is consistent with the structure of campaign metadata, where some creative descriptors may have been recorded only for specific platforms, campaign types, or historical periods.

The Google Analytics recency variable `days_since_last_ga_session_asof` also had high missingness, at 87.45%. This missingness is consistent with the construction of the variable: it can be observed only when a previous Google Analytics session is linked to the same client identifier before quiz completion. Missing values in this field may therefore indicate the absence of a prior observable session rather than a data-processing error.

Quiz and customer-profile variables generally had lower missingness, usually around 6–7%. This group included variables such as `monthly_budget`, `gender`, `has_computer`, and `internet_stability`. Their missingness likely reflects incomplete quiz responses, optional questions, or transformation issues in the quiz data pipeline.

Missingness by target class was relatively similar for most variables. The largest

observed differences were approximately 1–2 percentage points for most fields. This suggests that missingness was not strongly concentrated in either the visited or non-visited class, although high-missingness campaign fields still required careful treatment during preprocessing.

The second part of the data quality analysis examined timestamp consistency. The timestamp checks found no cases where quiz end time was earlier than quiz start time. Similarly, there were no cases where the last recorded booking time was earlier than the first recorded booking time. These results indicate that the basic internal ordering of quiz events and booking updates was consistent in the dataset.

However, inconsistencies appeared when quiz timestamps were compared with scheduled trial-class times. In 107 rows, or 1.88% of the dataset, the scheduled trial-class time occurred before quiz start. In 98 rows, or 1.72%, `gap_lead_to_mk_hours` was negative, and in 92 rows, or 1.62%, `days_to_mk` was negative. These cases may reflect timezone effects, late quiz completions after an earlier booking, CRM update order, rescheduling logic, or data integration issues.

Since the affected share was relatively small, these anomalies did not define the overall structure of the dataset. Nevertheless, they were important to identify because booking timing is directly related to the prediction task and could introduce misleading patterns if treated as ordinary valid values.

The final part of the data quality analysis concerned outliers. Outlier detection was performed using the interquartile range rule. The interpretation of outliers depended on the type of feature. For bounded or cyclical variables, such as booking hour, an IQR-based flag does not necessarily indicate an invalid observation. It may simply reflect that most bookings are concentrated within common business hours, making less frequent hours appear statistically unusual.

Among continuous or count-like variables, the highest outlier shares were observed for `days_to_mk` at 8.51%, `gap_lead_to_mk_hours` at 7.50%, `quiz_completion_time` at 6.50%, and `impressions` at 5.78%. These values are plausible for behavioural and campaign data. Some users may complete the quiz unusually quickly or slowly, some trial classes may be scheduled unusually far in advance, and advertising campaigns may differ substantially in accumulated impressions before the quiz completion cutoff.

Therefore, missing, anomalous, and extreme values were not treated as automatic deletion criteria. Instead, they informed the preprocessing strategy. Missing values required feature-specific treatment depending on whether they represented absence of information, non-applicability, or incomplete recording. Timing anomalies required particular attention because they could indicate invalid temporal relationships. Numerical outliers were reviewed but not automatically removed, since many of them could represent valid behavioural or campaign variation. This distinction was especially important because the modelling strategy included both linear and tree-based methods: linear models can be more sensitive to extreme numerical values, whereas tree-based

models are usually more robust but may still reflect inconsistent timing patterns in their splits.

Chapter 6

Methodology

Figure 6.1 provides a graphical summary of the analysis methodology used in this chapter. It outlines the workflow from data extraction and preprocessing to model training, feature selection, hyperparameter optimization, and final evaluation.

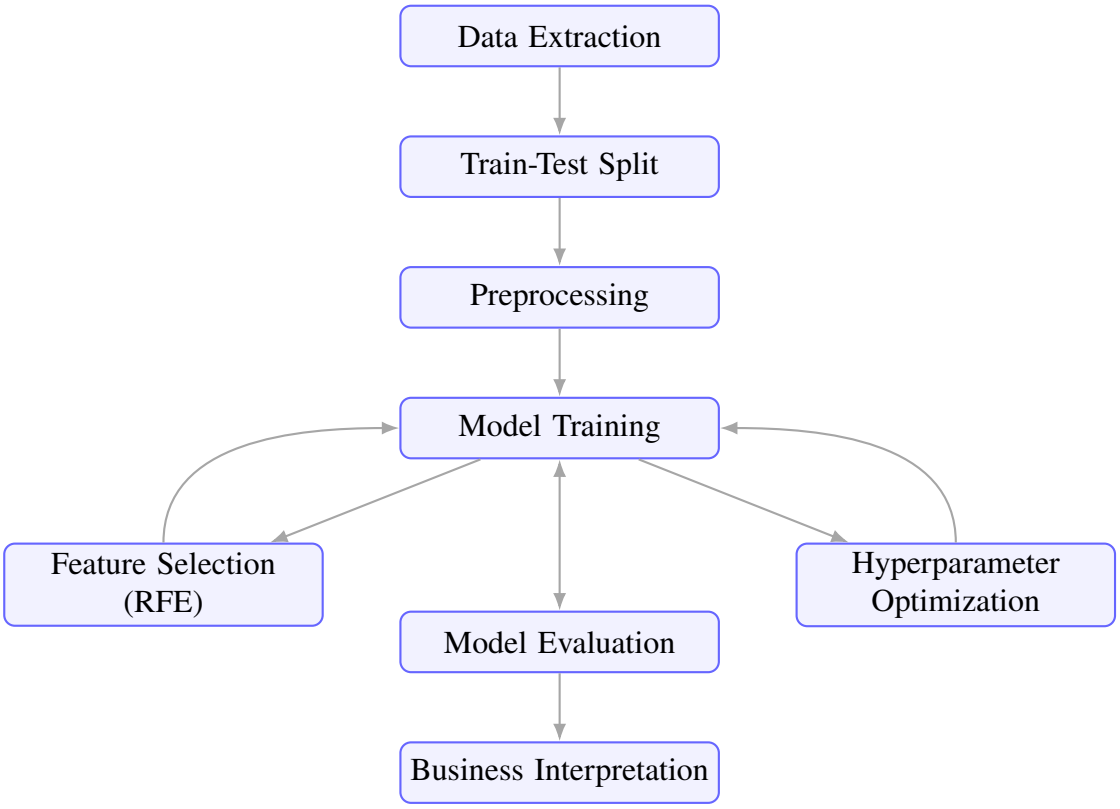

Figure 6.1: General overview of the analysis methodology

6.1 Lead Scoring: Problem Formalisation

The empirical task was formulated as a supervised binary classification problem. The objective was to estimate whether a lead–quiz-completion observation would result in trial-class attendance within a short-term operational horizon. The unit of observation was not a unique CRM lead, but a lead associated with a specific quiz completion, since each quiz completion represented a separate moment at which information became available for potential scoring.

Let the modelling dataset be defined as:

$$\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N}, \tag{6.1}$$

where $\mathbf{x}_i \in \mathbb{R}^m$ denotes the encoded and preprocessed feature vector for observation i, and y_i is the binary attendance outcome. The final target variable was defined as attendance within seven days after quiz completion:

$$y_i = \begin{cases} 1, & \text{if the lead attended a trial class within seven days after quiz completion,} \\ 0, & \text{otherwise.} \end{cases} \tag{6.2}$$

The seven-day target window was selected to align the modelling task with the operational use case. The initial version of the target distinguished between eventual attendance and non-attendance. However, exploratory analysis showed that most observed trial-class visits occurred within the first seven days after quiz completion. Later attendance cases were less directly aligned with near-term prioritisation, because they could be affected by later follow-up, rescheduling, or other post-quiz events. Therefore, the final target was defined as short-term attendance within seven days.

A central methodological constraint was the definition of the scoring moment. The model was intended to be applied after a lead completed the qualification quiz and before the attendance outcome became known. This corresponds to the moment when the company has already collected quiz, attribution, and booking-related information, but has not yet observed whether the lead will attend the trial class.

Let t_i^{score} denote the operational time at which a score is produced for observation i. In this project, the scoring moment is aligned with quiz completion:

$$t_i^{\text{score}} = t_i^{\text{quiz\_completion}}. \tag{6.3}$$

The qualification quiz and the booking step occur back-to-back in the same session, so the booking record, including the scheduled trial-class time, is complete at the scoring moment. This is why the booking-to-class interval (`gap_lead_to_mk_hours`) is an admissible feature: it is determined at booking, which precedes scoring, and does not draw on any information arising after the score is produced.

A candidate feature x_{ij} is admissible only if the information required to construct it is available no later than the scoring moment:

$$t(x_{ij}) \leq t_i^{\text{score}}. \tag{6.4}$$

This temporal-validity requirement distinguishes predictive information from information that is merely correlated with the outcome retrospectively. A variable recorded after the scoring point may be highly predictive because it already reflects part of the future business process. Its inclusion would create target leakage and could produce an unrealistically favourable offline evaluation. Therefore, the operational feature set had to reproduce the information that would be available when a new lead is scored.

The feature vector $\mathbf{x}_i$ contained information derived from CRM and booking records, quiz responses, Google Analytics data, campaign metadata, and aggregated client-level variables after preprocessing. However, variables were included in the predictive feature set only if they satisfied the temporal-validity requirement. Some fields were useful for target construction, data validation, or descriptive analysis, but were not appropriate as model inputs if they reflected information observed after quiz completion.

The model estimates the conditional probability of short-term attendance:

$$\widehat{p}_i = \widehat{P}\left(y_i = 1 \mid \mathbf{x}_i\right), \qquad 0 \leq \widehat{p}_i \leq 1. \tag{6.5}$$

The output is therefore a continuous attendance score rather than only a hard class label. This score supports two related objectives. The first is ranking: observations that are more likely to result in attendance should receive higher scores than observations that are less likely to attend. The second is probability estimation: predicted probabilities should correspond reasonably well to observed attendance frequencies. These objectives are related but not identical, since a model can rank leads effectively while still producing poorly calibrated probabilities [35].

For operational use, the continuous score may be transformed into priority groups:

$$\text{Segment}_i = \begin{cases} \text{Low}, & \widehat{p}_i < \tau_1, \\ \text{Medium}, & \tau_1 \leq \widehat{p}_i < \tau_2, \\ \text{High}, & \widehat{p}_i \geq \tau_2, \end{cases} \tag{6.6}$$

where τ_1 and τ_2 are selected score boundaries. These thresholds are not universal properties of the algorithm. They are operational parameters that should be selected using validation data, expected segment sizes, observed attendance rates, and the capacity of the teams responsible for lead follow-up. The resulting groups should therefore be interpreted as decision-support categories rather than natural classes in the population.

A conventional binary prediction can also be produced using a single threshold τ:

$$\widehat{y}_i(\tau) = \begin{cases} 1, & \widehat{p}_i \geq \tau, \\ 0, & \widehat{p}_i < \tau. \end{cases} \tag{6.7}$$

The default threshold of 0.5 has no special operational status. It may be unsuitable when attendance is the minority outcome or when the business can prioritise only a limited number of booked leads. Threshold selection should instead reflect the intended action and the relative cost of different errors.

At threshold τ, an abstract expected-cost function can be written as:

$$\text{Cost}(\tau) = c_{\text{FP}}\,\text{FP}(\tau) + c_{\text{FN}}\,\text{FN}(\tau), \tag{6.8}$$

where c_{FP} and c_{FN} represent the operational costs assigned to false-positive and false-negative decisions. A false positive may direct intensive follow-up toward a lead who ultimately does not attend. A false negative may place a future attender in a low-priority group. Even when these costs cannot be expressed precisely in monetary units, this formulation makes explicit that threshold choice depends on the intended business action.

Finally, the score is not interpreted as a causal estimate. A high predicted probability indicates that the observation has patterns associated with attendance under the fitted model. It does not establish why the individual will attend, nor does it imply that changing a single recorded feature would cause attendance. The model is designed to support prioritisation and resource allocation, while final operational decisions may incorporate additional business rules and human judgement.

As shown in Section 5.3.1, the dataset contained 5,690 observations with 23.08% positive cases, which motivated the evaluation strategy described below.

6.2 Train–Test Split

The dataset was divided into training and test subsets using a time-based split. This choice reflected the intended operational use of the model: the model would be trained on historical observations and then applied to future leads. A random split could mix earlier and later observations across both subsets and produce an overly optimistic estimate of performance if campaign composition, acquisition channels, or lead behaviour changed over time.

The split was based on the quiz start timestamp stored in `time_started`. The column was converted to datetime format, rows with invalid timestamps were removed from the split procedure, and the remaining observations were sorted chronologically. The first 80% of observations were assigned to the training set and the last 20% to the test set. The target variable was `visited_tc_7d_target`, which was separated from the feature matrix after the split.

The resulting training set contained 4,552 observations, while the test set contained 1,138 observations. The training period ended approximately on 11 February 2026 at 17:45:07 UTC. The test set contained later observations and extended until 12 March 2026 at 15:40:58 UTC.

This design produced a chronological evaluation setting: the model was fitted on earlier quiz completions and evaluated on later quiz completions. No random seed was required because the observations were not shuffled. The structure better approximated the production scenario, where future leads arrive after the historical data used for model development.

The split was performed before distribution-dependent preprocessing steps. Transformations that required estimating patterns from the data, such as rare-category handling, were fitted only on the training subset and then applied unchanged to the test subset. At this stage, hyperparameter tuning had not yet been performed, so the time-based test set was reserved for out-of-sample evaluation and was not used for model optimisation.

6.3 Data Preprocessing

Data preprocessing was performed after the time-based train–test split. This order was necessary because some preprocessing operations depended on the empirical distribution of the data. Estimating such operations on the full dataset before splitting would allow information from the later test period to influence the training pipeline. Therefore, distribution-dependent transformations were fitted only on the training subset and then applied unchanged to the test subset.

The purpose of preprocessing was to transform the analytical extract into a stable modelling dataset suitable for supervised binary classification. The main objectives were to reduce noise in categorical variables, handle missing values consistently, preserve information about missingness where it could be informative, remove raw datetime fields after extracting useful temporal features, and exclude technical identifiers that should not directly enter the model.

The first preprocessing step addressed rare categories in selected categorical variables. Several acquisition, course, and campaign-related fields contained values represented by only a small number of observations. Such categories can be unstable during model training because the model may learn patterns specific to a few historical cases rather than relationships that generalise to future leads. This issue is particularly relevant for marketing variables, where campaign names, source labels, and tracking parameters may change over time.

Rare-category handling was applied to selected categorical fields, including: source, course name, campaign offer, campaign event, utm source, and utm content. For each of these columns, category frequencies were calculated using the training data only. Categories with fewer than 50 observations in the training subset were replaced with the

shared label `Unknown`. The resulting mapping was then applied to the test subset, so rare categories in training and categories appearing only in the later test period were handled consistently.

Country was processed separately using the same logic but with a higher threshold. Countries represented by fewer than 100 observations in the training subset were replaced with `Unknown`. This reduced the risk of learning unstable market-specific patterns from countries represented by only a small number of observations. In methodological terms, this step did not remove acquisition or geographic information, but represented it at a more stable level of granularity.

The second preprocessing step handled missing values in categorical variables. Categorical columns were identified based on their data type, including `object`, `category`, and string-like columns. Missing values were detected both as formal null values and as empty strings. For each categorical column with missing values, a separate missingness indicator was created using the pattern:

`column_name_missing_flag`

After these indicators were created, the original missing categorical values were replaced with the shared label `Unknown`. This allowed the models to process categorical variables without dropping observations. The missingness indicators were included because the absence of a value may itself contain useful information. For example, a missing budget answer may reflect incomplete quiz engagement, while a missing attribution field may reflect tracking limitations or differences in the acquisition path. The missing flag preserved the distinction between an explicitly recorded unknown category and a value that was absent in the source data.

This approach avoided target-dependent imputation. Missing values were not filled using the target variable, target-class averages, or information observed after the prediction moment. Missingness was therefore preserved as a potential signal without allowing the outcome variable to influence feature construction.

The next preprocessing step removed raw datetime columns from the modelling data. Raw timestamps were not used directly as model inputs because many machine learning algorithms cannot process datetime objects without transformation. In addition, raw dates may encode calendar-specific artefacts that do not generalise to future observations. Instead, temporal information was represented through derived variables created earlier in the pipeline, such as weekday, booking hour, quiz completion duration, and the time gap between quiz completion and the scheduled trial class.

The final preprocessing step removed technical identifier columns. In particular, the pseudonymised CRM identifier `amocrm_id` was excluded from the predictive feature set. The identifier was useful for joining source tables, checking repeated quiz completions, and validating records, but it did not represent a behavioural or demographic characteristic of the lead. If retained, the model could learn accidental associations specific to historical CRM entities rather than generalisable patterns.

Overall, the preprocessing pipeline reduced categorical noise, preserved informative missingness, removed unsuitable raw timestamp fields, and excluded technical identifiers. These steps prepared the dataset for modelling under more realistic prediction conditions, where future leads may contain unseen categories, incomplete fields, and imperfectly recorded operational data.

6.4 Baseline: Historical Attendance Rate

Before training machine learning models, a simple no-model baseline was established. The baseline represented a situation in which no individual-level prediction was available. Every observation was assigned the same predicted probability equal to the historical trial-class attendance rate observed in the training data.

Let $\bar{y}_{train}$ denote the mean value of the target in the training set:

$$\bar{y}_{train} = \frac{1}{N_{train}} \sum_{i=1}^{N_{train}} y_i.$$

The baseline prediction for every observation was then:

$$\hat{p}_i^{baseline} = \bar{y}_{train}.$$

This baseline corresponds to using only the average historical conversion from lead to trial-class attendance within seven days. The historical attendance rate was calculated using Kodland database records from the beginning of 2026. Over this period, the baseline attendance rate for the overall trial cohort was 7%. For the Latam segment, the corresponding baseline was slightly higher at 8%, measured over the same period. These values were used as a simple historical benchmark for comparing observed trial attendance across segments and for identifying deviations from the expected attendance pattern [24].

The baseline has no ranking ability because all observations receive the same score. Its purpose is to provide a minimum reference point for evaluating whether machine learning models add predictive value. A useful model should improve upon this baseline by separating leads into groups with different observed attendance rates and by ranking actual attenders above non-attenders more effectively than chance.

From a business perspective, this baseline represents the expected situation without a predictive scoring model. In that case, the company can rely only on the aggregate attendance rate of the booked-lead population or on manual judgement. A machine learning model is justified only if it provides more informative segmentation than this aggregate historical rate.

6.5 Modelling

The modelling stage aimed to evaluate whether machine learning methods could improve lead ranking and attendance probability estimation relative to the historical attendance-rate baseline. Since the task was formulated as binary classification, candidate models were trained to estimate the probability of the positive class, corresponding to trial-class attendance within seven days.

The workflow followed the same general structure for all algorithms. The preprocessed training data were used to fit the model, and the fitted model then generated predicted probabilities for the later test observations. The test set was not used during model fitting or hyperparameter selection.

The candidate modelling approaches were selected to balance interpretability, robustness, and ability to capture nonlinear patterns. A linear model provided a transparent benchmark and tested whether a relatively simple additive relationship between predictors and attendance was sufficient. Tree-based models were included because they can capture nonlinear effects, threshold behaviour, and interactions between variables without requiring these relationships to be specified manually. This was relevant for the lead-attendance task because the effect of a variable may depend on context. For example, booking timing may matter differently across traffic sources, countries, or levels of technical readiness.

The models were trained to output probabilities rather than only hard class labels. This was important because the operational goal was lead prioritisation. A probability score allows leads to be ranked from lower to higher expected attendance likelihood and enables different decision thresholds to be selected depending on operational capacity.

Model assessment used the evaluation framework described in Section 4.2.4. The primary metrics were ROC-AUC for ranking quality and precision–recall analysis for minority-class retrieval, given that the positive class represented 23.08% of the dataset. Predicted scores were additionally analysed by priority groups to verify monotonic ordering of observed attendance rates across segments.

Following the initial modelling stage, the model with the best performance was selected for hyperparameter tuning. Therefore, the initial modelling results should be interpreted as a preliminary assessment of modelling feasibility rather than as the final optimised performance of each algorithm. Further performance improvements were explored through model-specific hyperparameter optimisation and the selection of operational thresholds aligned with business constraints.

6.5.1 Logistic Regression

Logistic Regression was used as one of the baseline classification models.

To ensure a leakage-safe preprocessing workflow, the transformations were implemented with a `Pipeline` and `ColumnTransformer` from scikit-learn. This setup

ensured that all preprocessing steps were fitted only on the training data and then applied unchanged to the test data. The features were split into numerical and categorical groups based on their data types.

Numerical features were imputed using the median strategy and then standardized with `StandardScaler`. Standardization was necessary because Logistic Regression is sensitive to feature scale, especially when predictors are measured on different ranges.

Categorical features were imputed using the most frequent category and then encoded with One-Hot Encoding. The encoder was configured with `handle_unknown="ignore"` to prevent errors when previously unseen categories appeared in the test set.

The preprocessing steps and the Logistic Regression estimator were combined into a single modeling pipeline. The classifier was trained with `class_weight="balanced"` to account for class imbalance in the target variable. The `lbfgs` solver was used, and the maximum number of iterations was increased to 2000 to support convergence.

6.5.2 Decision Tree

Decision Tree was used as a non-linear classification model to capture possible interaction effects between features. The model was trained on the same train-test split and target variable as the previous models, where the target variable was `visited_tc_7d_target`.

Preprocessing was implemented using a scikit-learn `Pipeline` together with a `ColumnTransformer`. Features were separated into numerical and categorical groups based on their data types. This setup ensured that all preprocessing steps were fitted only on the training data and then applied to the test data in a leakage-safe manner.

For numerical features, missing values were imputed using the median strategy. Unlike Logistic Regression, no scaling was applied, because Decision Tree models are not sensitive to the scale of numerical variables.

For categorical features, missing values were imputed using the most frequent category. After imputation, categorical variables were transformed using One-Hot Encoding. The encoder was configured with `handle_unknown="ignore"` to handle unseen categories in the test set without producing errors.

The preprocessing steps and the `DecisionTreeClassifier` were combined into a single modeling pipeline. The classifier was trained with `class_weight="balanced"` to account for class imbalance. To reduce overfitting and preserve interpretability, the maximum tree depth was limited and a minimum number of samples per leaf was specified.

6.5.3 Random Forest

Random Forest was used as a non-linear classification model to capture interaction effects and more complex decision boundaries than those represented by linear models. The model was implemented with `RandomForestClassifier` from scikit-learn and trained

on the same target variable, `visited_tc_7d_target`, and the same train-test split as the other models. This ensured that the performance comparison remained consistent across algorithms.

Before training, the feature set was divided into numerical and categorical variables based on their data types. Variables with types `object`, `category`, `string`, and `bool` were treated as categorical, while the remaining variables were treated as numerical. The preprocessing logic was implemented with a `ColumnTransformer`, which allowed different transformations to be applied to different feature groups within a single workflow.

Numerical features were imputed using `SimpleImputer(strategy="median")`. This means that missing values were replaced by the median value estimated from the training data. No scaling was applied to numerical variables, because Random Forest is a tree-based method and is not sensitive to differences in feature scale.

Categorical features were processed with a two-step pipeline. First, missing values were imputed using the most frequent category via `SimpleImputer`. Then, the categorical variables were transformed using One-Hot Encoding with option that previously unseen categories in the test set would not cause transformation errors.

All preprocessing steps and the classifier were combined into a single `Pipeline`. This setup ensured that transformations were fitted only on the training data and then applied consistently to the test data, which reduced the risk of data leakage. Hyperparameters of the Random Forest model were selected heuristically to provide a regularized baseline and reduce overfitting. The number of trees was set to 300 to ensure stable ensemble predictions, while the maximum tree depth was limited to 10 and the minimum number of samples per leaf was set to 20 to prevent overly complex trees and reduce sensitivity to rare category combinations. Class weights were balanced at the bootstrap-sample level, parallel computation was enabled across all available CPU cores, and a fixed random seed was used to ensure reproducibility.

After fitting, the model was used to produce both predicted classes and predicted probabilities for the positive class via `predict_proba`. The probability estimates were then used for ROC-AUC calculation. Feature importance was extracted from the fitted model using the built-in importance measure of the Random Forest estimator. Since categorical variables were one-hot encoded, importance values were assigned to individual encoded categories rather than to the original variables as a whole.

6.5.4 Gradient Boosting - Catboost

CatBoost was used as the gradient boosting model in this study. It was selected because it is well suited for tabular data with categorical variables and does not require manual One-Hot Encoding before training. Instead, categorical features are passed directly to the model, and their processing is handled internally by the algorithm.

The model was trained using `CatBoostClassifier` with loss function ”Logloss”, since the task was binary classification. During training, AUC was used as the evaluation metric. Early stopping was enabled to reduce the risk of overfitting and to avoid unnecessary boosting iterations. The parameter `od_type="Iter"` activated the overfitting detector, while `od_wait=50` specified that training should stop if the validation metric did not improve for 50 consecutive iterations. In addition, `use_best_model=True` ensured that the final model corresponded to the iteration with the best validation performance.

CatBoost is specifically designed to handle categorical features efficiently and is widely used in applied machine learning for tabular problems [15]. The original CatBoost papers describe both the ordered boosting framework and the built-in handling of categorical variables as core advantages of the method over conventional gradient boosting implementations[37].

6.5.5 Gradient Boosting - LightGBM

LightGBM was used as an additional gradient boosting model in this study. It was selected because it is an efficient implementation of gradient boosting decision trees and is well suited for tabular data with many features. In comparison with simpler tree-based models, LightGBM can capture more complex patterns while remaining computationally efficient.

Categorical features were not one-hot encoded. Instead, LightGBM's native support for categorical variables was used directly. Missing values in categorical columns were replaced with `"Unknown"`, after which the values were converted to the categorical data type. The list of categorical columns was then passed to the model through the `categorical_feature` argument.

Datetime features were converted into numerical form by transforming each date value into the number of seconds since 1 January 1970. This allowed the model to use temporal information as regular numeric input. Numerical features remained in their original format, and no additional scaling was applied because tree-based models are not sensitive to feature scale.

Early stopping was enabled through a stopping rule with 50 rounds. If the validation metric did not improve for 50 consecutive iterations, training was stopped. This prevented unnecessary boosting iterations and helped limit overfitting. In addition, training progress was logged every 100 iterations. After fitting, the model produced both predicted classes and predicted probabilities for the positive class via `predict_proba`, which were then used for ROC-AUC calculation.

6.6 Recursive Feature Elimination

Based on the comparison of the candidate models, LightGBM was selected as the main model for further tuning and feature selection. Its ranking performance was on par with the other strong candidates, and it was chosen on practical grounds: native handling of missing values and categorical features, efficient training, and a rich set of regularisation and class-imbalance parameters, rather than because it ranked leads better.

Since RFE requires repeated model fitting on progressively smaller subsets of features, a separate feature matrix was prepared for this procedure.

LightGBM was used as the base estimator inside the RFE procedure. This choice was made because LightGBM provides feature importance values and is efficient for tabular classification tasks. Instead of selecting a fixed number of features manually, several feature subset sizes were tested: 20, 30, 40, and 50 features. For each option, RFE recursively removed the least important features according to the LightGBM estimator.

To choose the final number of features, the training data was additionally split into training and validation parts. For each selected feature subset, a separate LightGBM model was trained and evaluated on the validation part. The tested subsets were compared using validation ROC-AUC, with validation F1-score used as an additional criterion in case of similar ROC-AUC values. The feature subset with the best validation performance was then selected for the final LightGBM + RFE model. This final model was trained only on the selected features and then evaluated on the untouched test set.

This setup ensured that the reduced feature set was evaluated under the same training protocol as the full model. In the next stage, this reduced representation was used to continue model tuning on a more compact and informative feature set.

6.7 Hyperparameter Optimization

Having retained LightGBM as the strongest candidate, this stage examined whether adjusting the learning procedure, with the set of input signals held fixed, could improve on the untuned model, which had already reached a ROC-AUC of 0.70 on the test set (see the Chapter 7, section Comparative Model Performance). Tuning proceeded in two stages: a broad random sampling of settings to locate a promising region, followed by a narrower, systematic search within that region. Whether either stage improved on the baseline is assessed in the Results chapter.

The tuning targeted a group of settings that together control how elaborate the model is allowed to become, since this governs the balance between fitting the historical data closely and generalising to new leads. A model permitted to grow too elaborate tends to memorise incidental noise rather than the patterns that genuinely separate leads who attend from those who do not. These settings fall into four roles: how the model is assembled (the number of decision trees and the size of each correction they add),

how detailed each individual tree may be (its depth, its number of end-points, and the minimum number of leads behind each split), how much deliberate variation is introduced between trees (the share of leads and of signals each tree is trained on), and how strongly unnecessary complexity is penalised.

Three choices were kept constant across both stages. First, each candidate configuration was judged by ROC-AUC rather than plain accuracy: because far fewer leads attend than do not, accuracy can be high simply by predicting that almost no one attends, so it is an unreliable measure under class imbalance, whereas ROC-AUC reflects how well the model orders attendees above non-attendees [20]. Consistent with this, the two outcomes were weighted equally during training. Second, configurations were compared using cross-validation that respected the order of events: the data were split by time, so each configuration was always evaluated on leads that came after those it learned from. This prevents a configuration from being rewarded for using information that would not yet exist when a real lead is scored - the same concern with the timing of information that constrains the choice of input signals throughout this work [4]. Third, once a best configuration was chosen, the final model was fitted on the earlier 80 % of the training data while the most recent 20 % was held back to decide when to stop training; training halted once performance on this recent portion ceased to improve. This portion is internal to training and separate from the held-out test set used for final evaluation.

6.7.1 Random Search

The first stage explored a broad range of settings. The search was allowed to use between 150 and 350 trees, each adding a small correction (a step size between 0.03 and 0.08); to keep trees shallow (four to eight levels deep, with 15 to 63 end-points) and based on at least 20 to 80 leads per split; to train each tree on a random 70–90 % of the leads and signals; and to apply a complexity penalty ranging from none up to a moderate level. Because specifying a range for each of the nine settings yields far too many combinations to test exhaustively, `RandomizedSearchCV` drew ten combinations at random. This reflects the established finding that, under a fixed computational budget, random sampling explores a large space of settings more efficiently than checking every point on a grid [5]. The configuration that ranked leads best was retained.

6.7.2 Grid Search

The second stage refined the result of the first. Rather than sampling widely, a narrow grid of candidate values was placed around the configuration identified by the random search, and `GridSearchCV` trained the model on every combination within it. The aim was no longer to explore the space but to check whether a small, systematic adjustment near the most promising region could yield a better configuration than the one random sampling had happened to land on. The evaluation was otherwise identical to the first

stage, the same time-ordered cross-validation, the same ROC-AUC criterion, and the same time-based early-stopping refit, so that the two searches could be compared on equal terms. The resulting model was added to the overall comparison in Table 7.1; its performance, together with a comparison of training and test ROC-AUC and the most influential input signals, is reported in the Results chapter.

6.8 Cross-Validation Procedure

The stability of the model's performance was assessed with five-fold cross-validation. The aim was not to tune the model further but to check whether its out-of-sample performance was consistent across different periods, rather than an artefact of one particular train/test split.

As in the earlier tuning stages, the folds were generated with `TimeSeriesSplit`, which preserves the chronological order of the data: each fold trains on an initial span of leads and validates on the period immediately following it, so the model is never validated on leads that precede those it learned from. This keeps the evaluation consistent with the timing constraint applied throughout this work and avoids crediting the model with information it could not have at scoring time [4]. For every fold, accuracy, precision, recall, F1-score, and ROC-AUC were recorded on both the training and the validation portion, with the two classes weighted equally during training. The per-fold validation ROC-AUC, together with its mean across folds, is the quantity of primary interest, since it summarises how stably the model ranks leads on data it has not seen.

6.9 Model Evaluation

A single aggregate metric says little about whether the model is useful in the operational setting it was built for. This section therefore evaluates the selected LightGBM model in the terms that matter for lead prioritisation: whether its score separates leads into groups with clearly different attendance rates, and how much benefit targeting the higher-scoring groups would bring relative to treating all leads equally.

6.9.1 Score Segmentation Strategy for Business

A continuous predicted probability is difficult for non-technical teams to act on directly. Therefore, the model score was transformed into three operational segments: low, medium, and high. This segmentation was designed to make the model output easier to interpret and to support lead prioritisation in a business setting.

The bucket boundaries were defined using two fixed thresholds based on the terciles of the score distribution in the training data. The resulting thresholds were 0.275 and

0.532, producing three groups of approximately equal size in the training set, with about 1,518 leads in each bucket. Scores below 0.275 were assigned to the low bucket, scores between 0.275 and 0.532 to the medium bucket, and scores above 0.532 to the high bucket.

Importantly, the thresholds were estimated only on the training data and then kept fixed for validation and operational use. This avoids re-defining the meaning of a "high" or "low" lead based on each new batch of data. As a result, the segmentation remains stable over time and independent of future observations. In production, new leads can therefore be assigned to one of the three buckets using the same fixed score boundaries.

6.9.2 Business-Oriented Evaluation (Top-K, Lift)

The model was also evaluated using business-oriented ranking metrics. In this context, the most relevant question is whether the model can rank leads in such a way that a relatively small share of the highest-scoring leads contains a disproportionately large share of eventual attendees.

For this purpose, the score buckets were evaluated using a top-K and lift framework. The top-K perspective measures how many actual attendees are captured when only the highest-scoring share of leads is prioritised. In the three-bucket segmentation used in this work, the high bucket can be interpreted as the top segment of leads by predicted attendance probability, while the combination of the high and medium buckets represents a broader prioritisation strategy with larger coverage.

For each bucket, the following quantities were computed on the held-out test set: the number of leads in the bucket, the share of all leads assigned to the bucket, the share of all actual attendees captured by the bucket, the bucket-level attendance rate, and the lift relative to the overall test-set attendance rate. The attendance rate of a bucket was calculated as:

$$\text{Attendance Rate}_b = \frac{\text{Number of attendees in bucket } b}{\text{Number of leads in bucket } b}.$$

The lift of a bucket was then calculated as the ratio between the bucket's attendance rate and the overall attendance rate in the test population:

$$\text{Lift}_b = \frac{\text{Attendance Rate}_b}{\text{Overall Attendance Rate}}.$$

A lift greater than one indicates that leads in the bucket attend more often than the average scored lead, while a lift below one indicates that they attend less often than average. Therefore, lift provides an intuitive measure of how concentrated high-value leads are within each score segment.

This evaluation is directly connected to the business use case. If the highest-scoring bucket captures a large share of eventual attendees while containing a smaller share of

total leads, the model can help concentrate call-centre effort where attendance is most likely. Conversely, if a low-scoring bucket contains few eventual attendees, it can be assigned to a lower-intensity nurturing process. Thus, the top-K and lift analysis translates model ranking quality into operational terms: coverage of attendees, prioritisation efficiency, and potential resource allocation value.

Importantly, these metrics should be interpreted within the population that the model actually scores. In this work, the model is applied to leads who have completed the qualification quiz. Therefore, lift measures the model's ability to rank leads within this qualified population, rather than the effect of the qualification process itself. The resulting figures should be understood as projected prioritisation gains based on observed test-set composition, not as causal evidence that acting on the score increases attendance.

6.9.3 Interpretability Analysis Setup (SHAP)

A score that drives operational decisions has to be explainable, not only accurate: the call-centre and marketing teams using the buckets need some account of why a lead is rated high or low, and the analyst maintaining the model needs to confirm that it relies on sensible, legitimately available signals rather than artefacts. Because the model is a gradient-boosted tree ensemble whose internal structure is not directly readable, its predictions were examined using SHAP (SHapley Additive exPlanations), a method that attributes each prediction to its input features by distributing the prediction among them according to a game-theoretic notion of fair contribution [31]. The attributions are additive and locally faithful, which means the contributions assigned to the features of a given lead sum to that lead's score and can be aggregated across leads to describe the model's behaviour as a whole.

The attributions were computed with `TreeExplainer`, an exact and efficient implementation of SHAP designed specifically for tree ensembles [29]. It was applied to the trained LightGBM model on a random sample of 1,000 leads drawn from the test set, a sample large enough to characterise the model's behaviour while keeping the computation and the resulting plots readable. Attributions were taken with respect to the positive class (attendance of the trial class), so that a positive SHAP value indicates that a feature pushed the model towards predicting attendance and a negative value that it pushed against it. Global feature importance was then summarised as the mean absolute SHAP value of each feature across the sampled leads, which measures how much each feature moved the prediction on average, regardless of direction.

Chapter 7

Results

7.1 Comparative Model Performance

In figure 7.1, ROC-AUC for both train and test for logistic regression is illustrated. The logistic regression model shows modest discrimination, with Train AUC = 0.70 and Test AUC = 0.69, which places performance around the acceptable-to-poor boundary and suggests the model generalizes similarly from train to test with little sign of overfitting. In practical terms, it can rank positive cases above negative ones about 70% of the time.

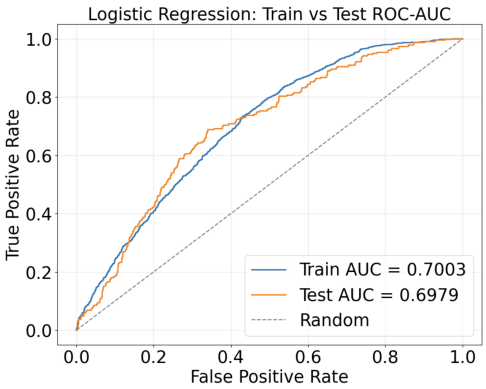

Figure 7.1: ROC-AUC: logistic regression

The decision tree model 7.2 exhibits modest discriminatory ability, with a Train AUC of 0.66 and a Test AUC of 0.63, indicating performance in the poor range and only limited separation between the positive and negative classes. The relatively small gap between training and test performance suggests that the model does not strongly overfit.

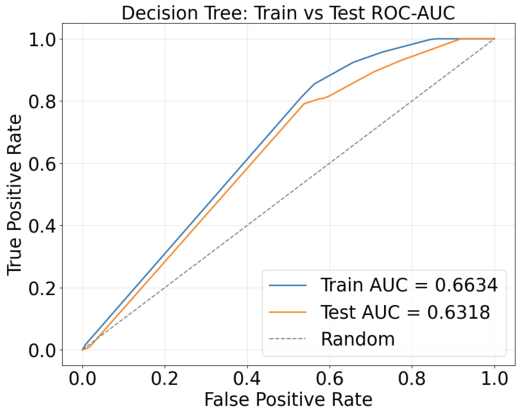

Figure 7.2: ROC-AUC: decision tree

In figure 7.3, ROC-AUC for both train and test for random forest is illustrated. The random forest model demonstrates comparatively stronger in-sample discrimination, with a Train AUC of 0.81, which falls in the good range, while the Test AUC of 0.69 indicates only average out-of-sample performance and a noticeable decline on unseen data. This gap between training and test AUC suggests some degree of overfitting, meaning that the model captures patterns in the training data more effectively than it generalizes to new observations.

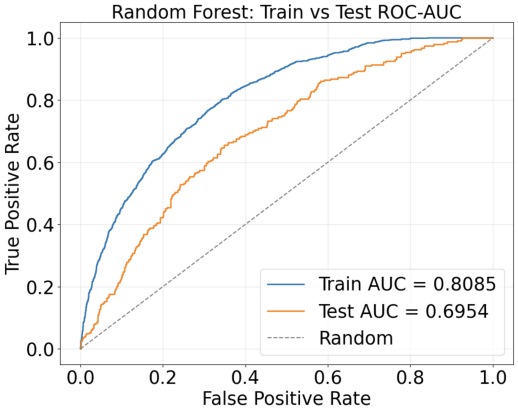

Figure 7.3: ROC-AUC: Random Forest

The CatBoost model 7.4 demonstrates moderate discriminatory performance, with a ROC-AUC of 0.75 on the training set and 0.68 on the test set. The test ROC-AUC remains clearly above the random baseline of 0.5, which indicates that the model captures a meaningful signal and is able to rank attending leads above non-attending leads better than chance.

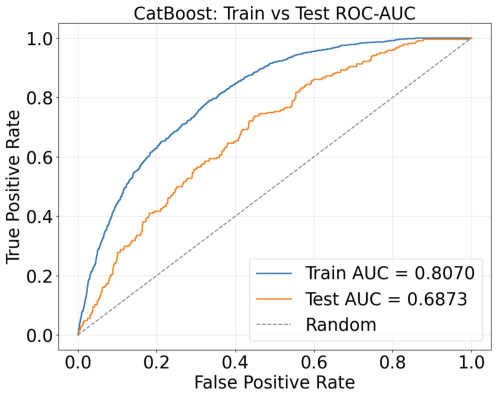

Figure 7.4: ROC-AUC: Catboost

The LightGBM model 7.5 achieved a ROC-AUC of 0.91 on the training set and 0.70 on the test set. The large gap between train and test ROC-AUC suggests that the initial LightGBM configuration overfits the training data. The model learns the historical training observations very well, but part of this learned structure does not generalise to later observations in the time-based test set. This gap may reflect excessive model complexity, temporal changes in traffic and campaign composition, or both.

Despite this limitation, the test performance remains competitive relative to the other initial models. Therefore, LightGBM can be considered a promising candidate for further hyperparameter optimisation. In particular, tuning parameters that control tree complexity, learning rate, number of boosting iterations, regularisation strength, and sampling may help reduce the train–test gap and improve generalisation to future leads.

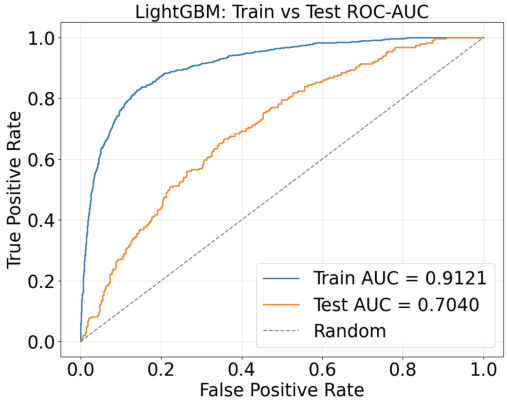

Figure 7.5: ROC-AUC: Light GBM

The initial models were close in ranking ability, and the apparent differences in accuracy, precision, and recall reflect where each model places its default decision

threshold rather than a genuine gap in discriminative power (provided in the Table 7.1). LightGBM was selected for further tuning and feature selection on practical rather than accuracy grounds: it is a flexible gradient-boosting model with native handling of missing values and categorical features, fast batch inference, and several regularisation and class-imbalance parameters that suit the subsequent optimisation stage. Logistic Regression was retained as the main linear benchmark.

The tree-based models did fit the training data more closely, reflected in larger train–test gaps, but this in-sample flexibility did not translate into better out-of-sample ranking, as the consolidated comparison below makes clear.

Table 7.1 shows that the choice of algorithm and the subsequent tuning had little effect on the model's ability to rank leads by attendance likelihood. ROC-AUC stays within a narrow 0.63–0.70 band across all eight variants, with every model except the single decision tree clustering between 0.68 and 0.70. The untuned LightGBM, at 0.70, was already among the best; neither feature selection (RFE), random search, nor grid search improved on it, and grid search was in fact marginally lower at 0.68. The accuracy, precision, and recall columns mainly reflect where each model places its decision threshold rather than any real difference in discriminative power: models with higher recall on the attending class (such as the decision tree at 0.81) simply flag more leads as likely attendees, at the cost of lower precision. Taken together, these results suggest that the ceiling on performance is set by the available data and input signals rather than by the choice or tuning of the algorithm, which shifts the model's practical contribution from raw predictive accuracy to how its probability output is grouped and used for lead prioritisation (Section 6.9.1).

Table 7.1: Test-set performance of the candidate models and the LightGBM variants. Precision, recall, and F1 refer to the attending (positive) class.

Model	**Accuracy**	**Precision**	**Recall**	**F1**	**ROC-AUC**
Logistic Regression	0.59	0.31	0.74	0.43	0.70
Decision Tree	0.50	0.27	0.81	0.41	0.63
Random Forest	0.58	0.30	0.74	0.43	0.70
CatBoost	0.65	0.33	0.59	0.42	0.69
LightGBM	0.67	0.34	0.57	0.43	0.70
LightGBM + RFE	0.62	0.32	0.70	0.44	0.70
LightGBM (random search)	0.65	0.33	0.61	0.42	0.69
LightGBM (grid search)	0.59	0.31	0.74	0.43	0.68
LightGBM + RFE (tuned)	0.58	0.30	0.74	0.43	0.71

7.2 Feature Selection Results

Among the tested feature subset sizes, the best validation performance was achieved with 40 selected features. This configuration obtained the highest validation ROC-AUC of 0.68 and was therefore used for the final LightGBM + RFE model. The comparison across subset sizes showed that reducing the feature space did not always improve model quality. The 20-feature subset performed worst, with a validation ROC-AUC of 0.64 and an F1-score of 0.44. Performance improved as the number of selected features increased: the 30-feature subset reached a ROC-AUC of 0.67 and an F1-score of 0.49, while the 50-feature subset achieved the highest validation recall at 0.79 and an F1-score of 0.50, but its ROC-AUC remained slightly below the 40-feature configuration. Overall, the 40-feature subset represented the best balance between model compactness and validation ranking quality.

Compared with the full LightGBM model, the LightGBM + RFE model changed the test metrics as follows: accuracy decreased by 0.05, precision decreased by 0.02, recall increased by 0.13, F1-score increased by 0.01, and ROC-AUC remained unchanged. The increase in recall means that the RFE-based model identified a larger share of users who actually visited the trial class. However, the decrease in accuracy and precision indicates that this improvement came at the cost of more false positive predictions. At the same time, ROC-AUC stayed at 0.70, which suggests that the overall ranking ability of the model was preserved after feature selection.

Overall, the RFE experiment showed that a smaller subset of features could maintain the same ROC-AUC as the full LightGBM model while slightly improving F1-score and substantially improving recall. However, this came with lower accuracy and precision. Therefore, the LightGBM + RFE model is preferable if the business priority is to identify more potential trial class visitors, while the full LightGBM model remains preferable if precision and overall accuracy are more important.

7.3 Hyperparameter Optimization Results

The random search returned a best cross-validated ROC-AUC of 0.65, with the configuration in Table 7.2. On the held-out test set the tuned model reached a ROC-AUC of 0.69 - effectively equal to the untuned LightGBM baseline of 0.70 (Table 7.1). Tuning therefore did not improve the model's ability to rank leads by attendance likelihood; its value was to confirm that the baseline was already close to the best this algorithm could achieve on the available data. The small gap between the cross-validated and test scores gives no sign of substantial over-fitting.

At the default threshold on the predicted probability (0.5), the model identified 148 of 244 actual attendees (recall 0.61) but raised many false alarms: 305 of the 894 non-attending leads were also flagged, leaving a precision of 0.33. This broad coverage bought

Table 7.2: LightGBM configuration selected by random search.

Hyperparameter	Value
n_estimators	150
learning_rate	0.08
max_depth	4
num_leaves	31
min_child_samples	20
subsample	0.8
colsample_bytree	0.7
reg_alpha	0.5
reg_lambda	1.0

with many false positives follows directly from weighting the two outcomes equally in training, which favours recall on the rarer attending class [20]. These threshold figures, however, matter less than how the probabilities order leads overall, since the model is meant to feed graded low/medium/high categories rather than a single hard cut-off. The test ROC-AUC of 0.69, above the 0.50 of random ordering, shows a modest but real separation; its practical usefulness is examined through the bucket-level attendance rates in Section 6.9.1.

7.4 Cross-Validation Results

The validation ROC-AUC was stable across all five folds, ranging only from 0.614 to 0.652 around a mean of 0.633 (Figure 7.6, Table 7.3). The narrowness of this band is a key finding: the model's ability to order leads by attendance likelihood does not depend heavily on which time period it is evaluated on. This consistency provides confidence that the holdout test ROC-AUC of approximately 0.70, reported separately as the final out-of-sample evaluation, reflects stable behaviour rather than a fortunate split. The difference between the CV mean and the holdout score is expected because the cross-validation uses expanding time-series folds: early folds train on smaller historical subsets and validate on later periods, making the estimate more conservative and sensitive to temporal drift. The slight upward drift in performance across folds is consistent with later folds having more training history available. This also suggests that, as additional labelled data becomes available, the final model performance may improve further, supporting the interpretation that the current ROC-AUC ceiling is partly driven by limited data availability rather than by an inherent absence of predictive signal.

A second pattern is: across every fold, the model fits the training data almost perfectly while its validation performance settles around 0.63, leaving a wide gap between the two. This near-perfect training performance indicates that the high-capacity configuration

Table 7.3: Validation ROC-AUC by cross-validation fold.

Fold	Validation ROC-AUC
1	0.614
2	0.633
3	0.622
4	0.644
5	0.652
Mean	**0.633**

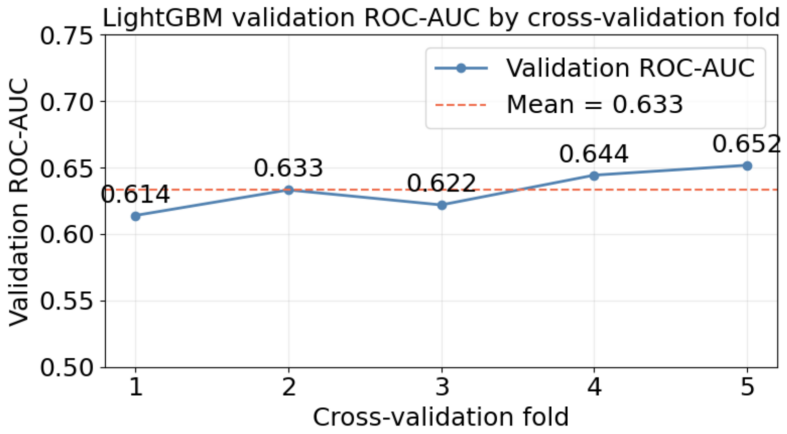

Figure 7.6: LightGBM validation ROC-AUC across the five cross-validation folds, with the mean shown as a dashed line.

adapts very closely to the training period, and a substantial part of that fit does not carry over to unseen leads. Such behaviour is expected for a gradient-boosting model with many trees, which is why regularisation, early stopping, and time-ordered validation were utilised. The validation ROC-AUC is therefore treated as the most reliable estimate of temporal generalization.

While additional Recursive Feature Elimination (RFE) and hyperparameter tuning experiments were performed as robustness checks, the final deployed configuration remains the baseline LightGBM. It provides a strong ROC-AUC with a simpler implementation and stable score-bucket behaviour. The conclusion drawn from this analysis is narrow and specific: the model is stable across time periods at a modest level of ranking performance, and it is this stable validation behaviour, not the training fit, that supports its use in the scoring pipeline.

7.5 Feature Importance

For Logistic Regression, model interpretability was assessed using the estimated coefficients. Since numerical features were standardized before training, the absolute magnitude of the coefficients can be used to compare the relative influence of features on the model prediction. Positive coefficients increase the log-odds of the positive class, while negative coefficients decrease them. For categorical variables, coefficients were interpreted at the level of one-hot encoded categories. In figure 7.7 the graph of coefficients is illustrated.

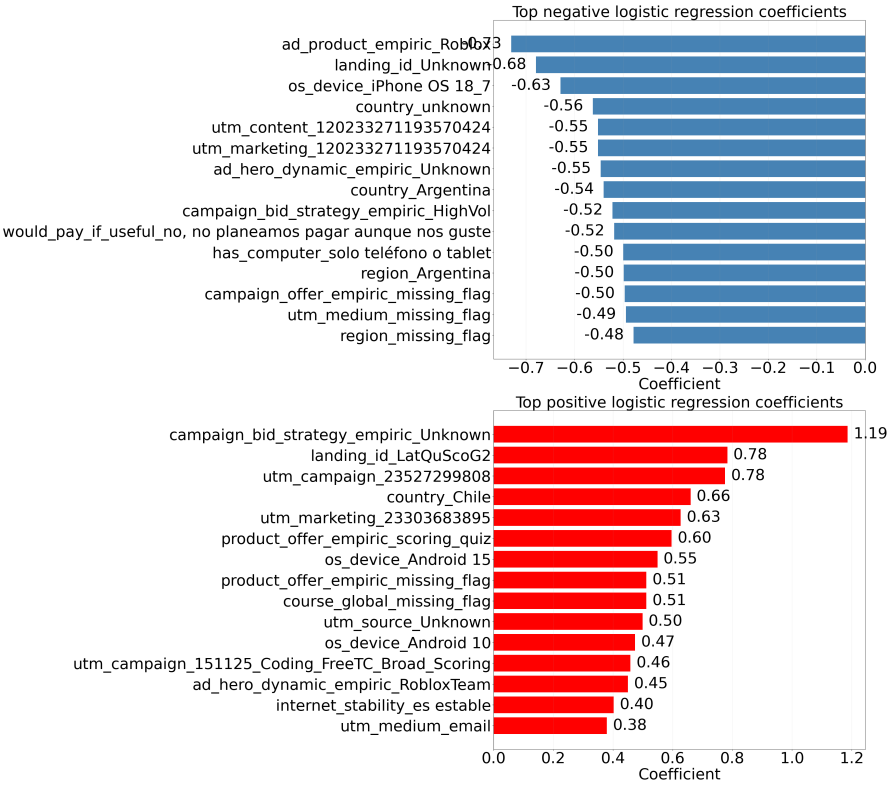

Figure 7.7: Logistic regression: Top-30 coefficients

The logistic regression indicates that campaign and landing-page attributes have the largest impact on trial-class attendance, while missing values and some device/region segments are associated with lower conversion probability.

For a Decision Tree, feature importance was obtained from the fitted Decision Tree model using the built-in `feature_importances_` attribute. Since categorical variables were one-hot encoded, importance values for categorical predictors are represented at the level of individual encoded categories rather than at the level of the original variable. In figure 7.8 the feature importances are demonstrated.

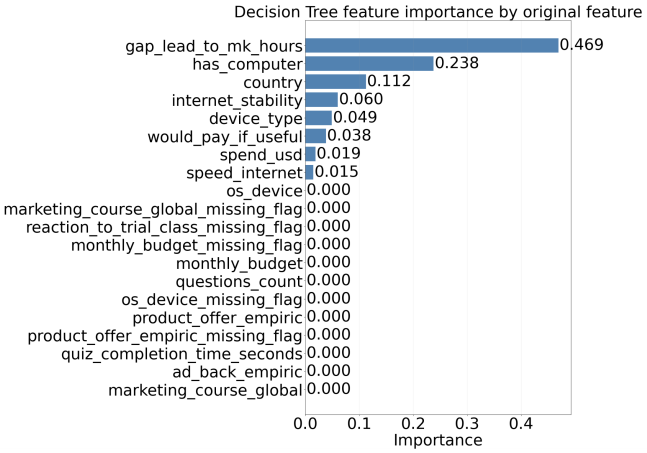

Figure 7.8: Decision Tree: Feature Importance

This pattern suggests that the decision tree captured the dependencies mainly through a small set of strong predictors, especially gap from lead creation to trial class and access to a computer. This is useful because it indicates that trial class attendance can be explained with a compact and interpretable rule set, but it also means the model may be sensitive to the dominant variables and less able to exploit weak signals from the rest of the feature set.

In figure 7.9 the feature importances for random forest model are demonstrated.

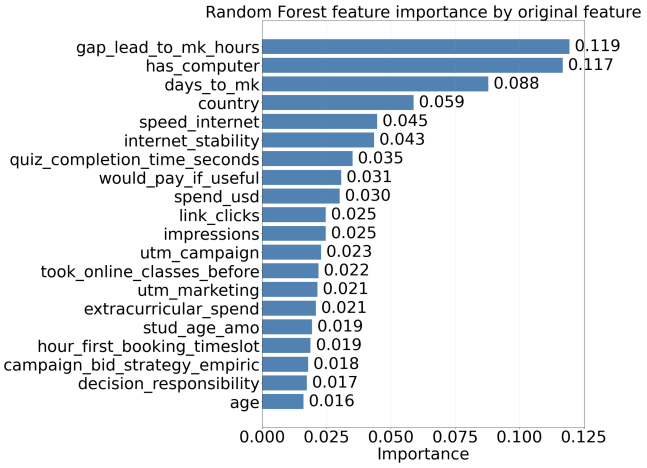

Figure 7.9: Random Forest: Feature Importance

The Random Forest importance plot shows that time-to-marketing variables dominate the model, with gap from lead creation to trial class, access to a computer and gap days to trial class as the strongest predictors. Secondary contributions come from country,

internet quality, and engagement-related variables such as quiz completion time, while marketing-channel and spending features have more moderate importance. Overall, the model appears to rely most on lead timing and access conditions rather than on campaign exposure alone.

Random Forest is less sparse and more robust than a single tree. While the strongest drivers of trial class attendance remain the same, the ensemble is able to extract additional predictive information from weaker features, which can improve generalization and provide a more balanced view of the underlying data.

In figure 7.10 the feature importances for Catboost model are demonstrated. Top features are similar to observed for Random Forest.

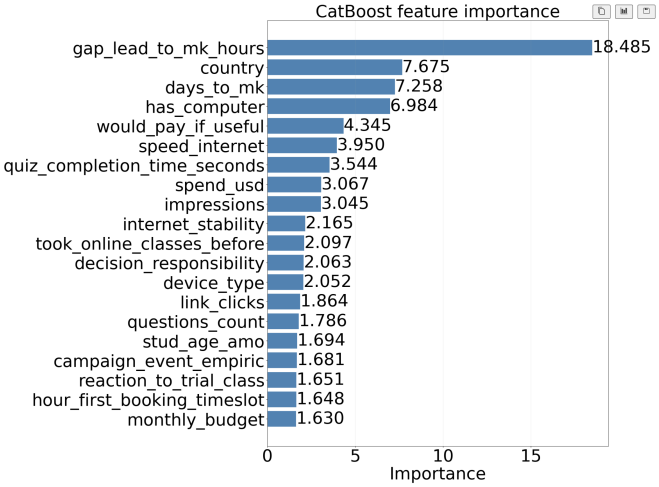

Figure 7.10: Catboost: Feature Importance

In figure 7.11 the feature importances for Light Gbm model are demonstrated. Top features are similar to observed for another forest esembles models.

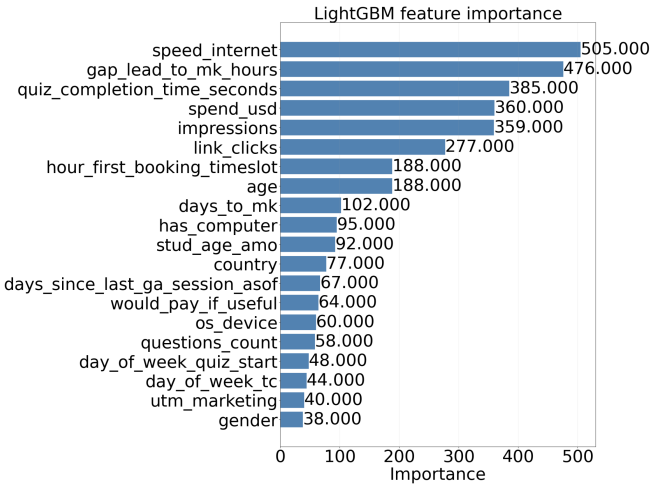

Figure 7.11: Light GBM: Feature Importance

7.6 SHAP Analysis: Drivers of Conversion

The resulting global ranking and the distribution of per-lead attributions are shown in the SHAP summary plot in Figure 7.12, in which features are ordered by mean absolute SHAP value and the colour of each point encodes whether the underlying feature value was high or low. Compared with standard variable-importance measures, SHAP has the additional advantage of providing a separate attribution for each feature and each individual record, rather than only a single aggregate importance score per feature. This setup supports two checks that are carried out in the following section: identifying which signals the model relies on most, and confirming that those signals are ones that are both behaviourally plausible and available at the moment of scoring. The latter check is essential here, since a feature may be highly predictive yet illegitimate if its value would not be known at quiz completion; interpretability is therefore used not only to explain the model but to audit it against the same information-timing constraint applied during feature construction.

Figure 7.12: SHAP summary plot for the LightGBM model on a sample of 1,000 test-set leads. Features are ordered by mean absolute SHAP value; colour indicates whether the feature value was high (red) or low (blue), and the horizontal position gives the direction and magnitude of each feature's effect on the predicted attendance probability.

The global SHAP ranking in Figure 7.12 shows that the model's predictions are driven by a small number of features. Three signals dominate: the interval between the lead and the booked trial class (`gap_lead_to_mk_hours`), whether the lead has access to a computer (`has_computer`), and the lead's country (`country`), followed by a second tier of quiz-derived signals such as reported internet speed, stated willingness to pay, and declared spending. The leading mean absolute SHAP values are listed in Table 7.4. The concentration of importance in a few features is consistent with the model's modest overall discrimination (test ROC-AUC around 0.70): the score reflects a handful of strong signals rather than a finely balanced combination of many weak ones.

Table 7.4: Leading features by mean absolute SHAP value on the sampled test-set leads.

Feature	**Mean |SHAP|**
gap_lead_to_mk_hours	0.42
has_computer	0.34
country	0.27
speed_internet	0.17
would_pay_if_useful	0.16
spend_usd	0.13
internet_stability	0.09
took_online_classes_before	0.07
age	0.07

To examine *how* the most influential continuous features affect the prediction, dependence plots for four of them are shown in Figure 7.13. Each point is one lead; the horizontal axis is the feature value and the vertical axis is that feature's SHAP contribution to the lead's predicted attendance probability. Throughout, these plots describe associations the model has learned, not causal effects: they show how the score responds to a feature, not how changing that feature would change actual attendance.

The interval between the lead and the booked trial class shows a clear, graded pattern. For short intervals, up to roughly 100 hours, the feature contributes positively, raising the predicted probability of attendance. Its contribution declines through the 100–170 hour range and becomes strongly negative beyond it, with the longest intervals pulling the prediction down sharply. The model has therefore learned that leads whose trial class falls soon after booking are more likely to attend, while long waits between booking and the class are associated with no-shows. This is the most important single pattern in the model and is operationally intuitive: a short, fixed interval gives less opportunity for intent to fade. Because this interval is determined at the moment of booking, which precedes scoring, it is available at prediction time and is a legitimate input rather than information about the future.

Reported internet speed shows an approximately monotonic, threshold-like relationship. Below a value of about three the contribution is slightly negative, and above it the contribution rises steadily, reaching its largest positive values at the highest speeds. For an online trial class this is intuitive: leads with better connectivity are more able to attend, and the model treats poor connectivity as a mild signal against attendance. Declared spending behaves less straightforwardly. Its contribution is close to neutral across low and middle values but turns negative at the high end, so that the model associates the largest declared spending with a slightly lower predicted probability of attendance. This pattern is not obviously interpretable and should be treated as a hypothesis rather than an established effect: one possible explanation is that leads who are already able and willing to pay may be less interested in attending a free trial class, whereas leads

with lower declared spending capacity may perceive the free trial as a more valuable opportunity and therefore be more likely to attend. Alternatively, the pattern may reflect an interaction with another feature, and it would benefit from validation with the business teams before being relied upon. Quiz completion time shows a penalty for haste: very fast completions (below roughly 100 seconds) contribute negatively, while moderate completion times contribute mildly positively, consistent with extremely rapid completion signalling low engagement. At long completion times the contributions become noisy, so little can be concluded there.

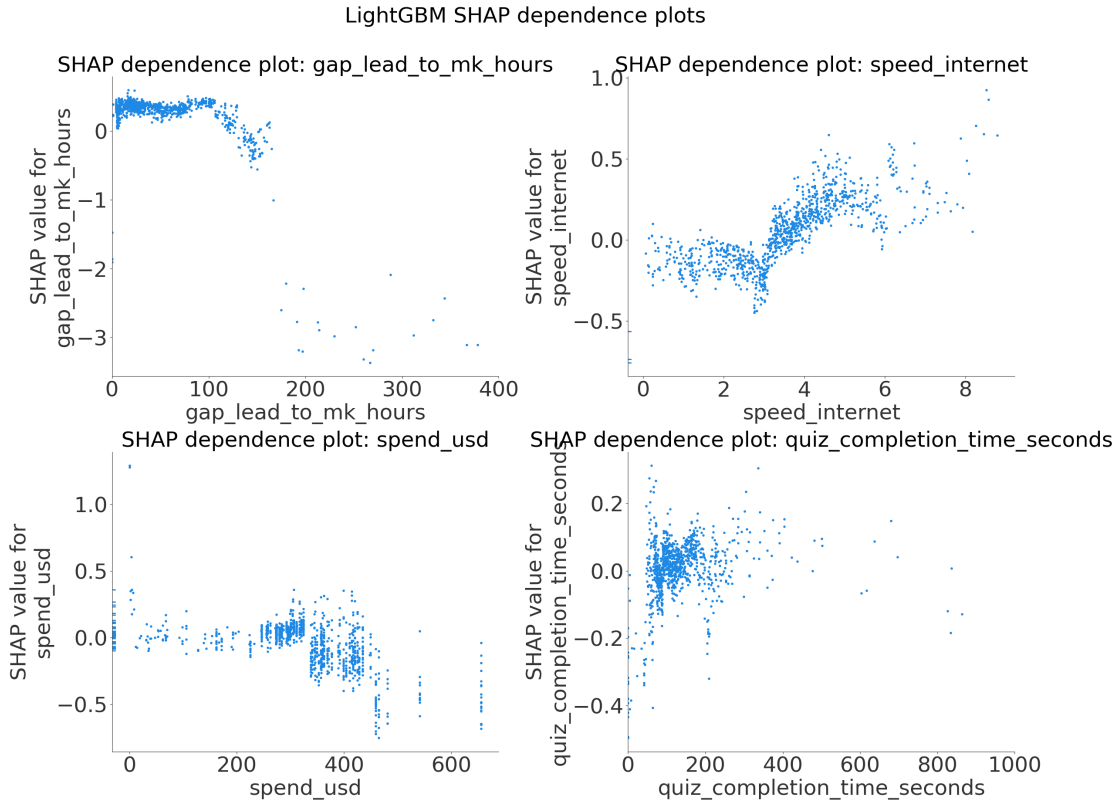

Figure 7.13: SHAP dependence plots for four leading continuous features. Each point is one sampled lead; the vertical axis gives the feature's contribution to that lead's predicted attendance probability.

Read together, these results also support the operational validity of the model. The signals it relies on most: the booking-to-class interval, connectivity and device access, country, and quiz answers, are all known at the moment the lead is scored, which is the first batch after a completed booking. None of them depends on information that arises only after the prediction is made. The interpretability analysis therefore serves two purposes at once: it makes the score explainable to the teams that act on it, and

it confirms that the model's most influential inputs are consistent with the information-timing constraint applied during feature construction.

7.7 Feature Importance Comparison

The SHAP ranking is broadly consistent with the LightGBM feature-importance plot, but the two diagnostics emphasize different aspects of the model. In the standard LightGBM feature importance, the most frequently used features are internet speed, gap hours to trial class, quiz completion time and campaign metadata. The SHAP ranking, however, places gap hours to trial class first, followed by computer question, country, internet speed and willigness to pay.

This difference is expected because LightGBM's built-in feature importance reflects how often a feature is used in tree splits, whereas mean absolute SHAP values measure the average magnitude of a feature's contribution to individual predictions. Therefore, variables such as `quiz_completion_time_seconds`, `impressions`, and `link_clicks` appear high in the LightGBM feature-importance ranking because they are used frequently in splits, but their average prediction impact is more modest in the SHAP analysis. Conversely, `has_computer` and `country` are less dominant in the LightGBM split-based ranking, but they have large SHAP values, suggesting that when these variables are used, they create stronger shifts in the predicted probability.

Overall, both methods point to the same substantive conclusion: the model is driven mainly by a compact group of timing, technical-readiness, geography, and quiz-response variables. The SHAP analysis provides the stronger interpretability evidence because it describes contribution size rather than only split frequency.

7.8 Business Impact and Error Analysis

7.8.1 Operational Value of the Segmentation

The operational value of the score segmentation was assessed by examining whether the low, medium, and high buckets corresponded to meaningfully different trial-class attendance rates. This validation was performed both on the training data and, more importantly, on the held-out test set. Table 7.5 reports the trial-class attendance rate within each score bucket. On both sets, the attendance rate increases monotonically from the low to the high bucket, which confirms that the model score orders leads in the intended direction rather than merely producing an arbitrary three-way split.

Table 7.5: Trial-class attendance rate by score bucket, on the training and test sets. Score ranges are the fixed training-based bucket boundaries.

Bucket	**Score range**	**Train Size**	**Train rate**	**Test Size**	**Test rate**
Low	[0, 0.275)	1518	1.5 %	308	8.1 %
Medium	[0.275, 0.532)	1517	9.2 %	479	19.0 %
High	[0.532, 1]	1517	59.8 %	351	36.5 %

Two observations follow the work. First, the separation is considerably sharper in train than in test: the high bucket attends at 59.8 % in training but only 36.5 % on the test set, and the gap between the lowest and highest buckets narrows correspondingly out of sample. This shrinkage is the expected counterpart of the modest test ROC-AUC and indicates that part of the in-sample separation reflects fitting of the training period rather than a pattern that fully generalises. Second, despite this shrinkage, the test gradient remains both monotonic and material: a lead in the high bucket attends roughly four and a half times as often as one in the low bucket (36.5 % against 8.1 %). For the purpose of ordering leads by priority, this out-of-sample gradient, not the steeper training one, is the relevant quantity, and it is large enough to be operationally useful.

The model does not change whether a booked lead attends; it reorders the effort spent on leads who have already booked. Its business value is therefore one of resource allocation rather than of conversion uplift. The test-set lift figures (Table 7.6) quantify this value directly. The high bucket contains roughly 31 % of booked leads but captures about 52 % of those who go on to attend, at an attendance rate 1.70 times the population average; the low bucket, by contrast, holds 27 % of leads but only about 10 % of attendees. Used as a prioritisation rule, this allows the call-centre team to contact the high- and medium-probability leads first and to route low-probability leads to a lower-intensity nurturing process (Section 8.3), so that limited follow-up capacity is concentrated where attendance is most likely.

The magnitude of this benefit depends on the operational constraint. When follow-up capacity is scarce relative to the number of bookings, ranking leads by attendance probability has clear value, because the same effort reaches a larger share of eventual attendees. When capacity is sufficient to contact every booked lead, the ranking is still useful for ordering and for differentiated treatment, but the gain is smaller.

It is important to state that these are projected efficiency gains derived from the observed bucket composition, not measured outcomes: the analysis shows how attendees are distributed across buckets, not that acting on the buckets causes more leads to attend. Establishing a causal effect on attendance would require a controlled experiment, which was outside the scope of this work and is discussed as a limitation.

7.8.2 Business-Oriented Results: Top-K and Lift

The business-oriented evaluation demonstrates that the model concentrates future attendees into the highest-scoring segments of the ranked lead population. Table 7.6 summarises the composition of each score bucket in the held-out test set, including the proportion of leads assigned to each bucket, the proportion of all attendees captured, the attendance rate, and the corresponding lift relative to the overall test attendance rate of 21.4 %.

Table 7.6: Test-set bucket composition

Bucket	Leads	% of leads	% of attendees	Attendance rate	Lift
Low	308	27.1 %	10.2 %	8.1 %	0.38
Medium	479	42.1 %	37.3 %	19.0 %	0.89
High	351	30.8 %	52.5 %	36.5 %	1.70
All	1138	100 %	100 %	21.4 %	1.00

The high bucket is the operationally important result. It contains about 31 % of all leads but captures roughly 52 % of those who eventually attend, at an attendance rate 1.70 times the population average. Read as a top-K targeting rule, this means that contacting only the highest-scoring third of leads would reach more than half of all future attendees. Extending coverage to include the medium bucket as well: the top 73 % of leads by score, captures close to 90 % of attendees, leaving only about 10 % of attendees in the low bucket. The low bucket, conversely, holds 27 % of leads but only one in ten attendees and attends at well below a third of the population rate (lift 0.38), which is what justifies routing it to lower-intensity nurturing rather than direct call-centre effort (Section 8.3).

These figures should be interpreted relative to the population the model actually scores, namely leads who have completed the qualification quiz. It is tempting to compare the high bucket's 36.5 % attendance against the historical attendance rate of roughly 7–8 % (8 % for the Latin American segment) recorded before the scoring approach was introduced. Such a comparison overstates the model's contribution, because the scored population already attends at 21.4 % overal, well above the historical rate, before any score is applied. Most of the gap between 8 % and the bucket rates therefore reflects the qualification step and the difference in populations, not the ranking produced by the model. The model's specific, defensible contribution is the separation it creates *within* the qualified population: concentrating attendees into the higher buckets and allowing the low bucket to be de-prioritised, as quantified by the lift column above.

7.8.3 Error Analysis and Failure Modes

Because the operating threshold favours recall on the minority attending class, the model's errors are dominated by false positives. On the test set it flagged 453 leads as likely attendees, of whom 148 in fact attended: 305 of the 894 non-attending leads were incorrectly flagged, while 96 of the 244 actual attendees were missed. In operational terms, acting on the model's positive predictions means spending follow-up effort on a substantial number of leads who will not attend. This is an acceptable trade-off when the cost of contacting a non-attendee is low relative to the cost of missing an attendee, but it does set a limit on how efficient the high-priority queue can be: roughly two in three leads flagged as likely attendees do not ultimately attend.

The errors are not uniformly distributed. The bucket analysis shows that the low bucket still contains a minority of genuine attendees: about 10 % of all attendees fall below the low threshold, so a strict policy of ignoring low-probability leads would forgo these. The model is also weakest in the medium bucket, where the attendance rate (19.0 %) is close to the population average and the score provides the least separation; leads in this band are the hardest to action on the basis of the score alone.

Two specific failure modes are worth noting for future work. First, the SHAP analysis surfaced a counterintuitive association in which the highest declared spending slightly lowered the predicted probability of attendance (Section 7.6). Whether this reflects a genuine behavioural pattern, aspirational quiz responses, or an interaction artefact is unresolved, and it is a candidate for review before the score is relied upon for high-spend segments. Second, the model's discrimination is modest in absolute terms (test ROC-AUC around 0.70), and the bucket gradient narrows out of sample relative to training. The model is therefore best understood as a coarse three-way prioritisation aid rather than a precise individual-level predictor of attendance, and its outputs should be used to order effort rather than to make irreversible decisions about individual leads.

Chapter 8

Operational Integration and Discussion

8.1 Interpretation of Findings

The results in the previous chapter point to a consistent and, in some respects, unexpected conclusion: the predictive ceiling for this problem is set by the available data rather than by the choice or tuning of the algorithm. Across eight model variants the test ROC-AUC remained within a narrow 0.68–0.70 band, and neither feature selection nor random and grid search improved on the untuned LightGBM baseline. The cross-validation analysis showed that this modest level of performance is at least stable: the validation ROC-AUC varied little across time-ordered folds, indicating that the model's ranking ability does not depend on a particular split. The honest reading of these findings is that no further gain was available from algorithmic refinement, and that effort spent searching for a markedly stronger model on the same features would have had a low expected return.

This reframes what the model contributes. A test ROC-AUC around 0.70 describes a model that ranks a randomly chosen attendee above a randomly chosen non-attendee about seventy percent of the time: useful for ordering leads, but far from a precise individual-level prediction. The value of the work therefore does not lie in the raw score but in what the score is turned into. Grouping the continuous probability into low, medium, and high buckets converts a modest ranking signal into an operationally meaningful one: on the held-out test set the high bucket attended at roughly 1.7 times the population rate and captured about half of all attendees, while the low bucket concentrated leads who attended well below average. A model that is mediocre as a point predictor can still be effective as a triage device, and it is in this triage role, separating booked leads into groups that warrant different levels of effort, that the model earns its place.

The interpretability analysis reinforces this reading and adds a second layer of confidence. The features the model relies on most are behaviourally coherent: the interval between booking and the trial class, access to a computer, country, connectivity, and a

small set of quiz responses. The dominant signal, a shorter interval between booking and the scheduled class being associated with higher attendance, matches the intuition that intent decays over time, and it is a signal known at the moment of scoring rather than one that leaks future information. Equally, the analysis surfaced a counterintuitive association around declared spending that remains unexplained and should be treated with caution. Taken together, the SHAP results suggest that the model has learned a small number of sensible, legitimately available patterns rather than spurious correlations, which matters more for a model intended to inform real decisions than a marginally higher accuracy figure would.

It is equally important to be clear about what the findings do not show. The model estimates the probability that an already-booked lead attends; it does not, and cannot on its own, increase attendance. The bucket-level differences describe how attendees are distributed across scores, not a causal effect of acting on those scores. As an additional empirical check, attendance was compared before and after the model was introduced into the operational process. Because only a limited observation window was available, this comparison was restricted to Latin America and covered one month before and one month after the intervention. The attendance rate increased from 8.5 % before the intervention to 9.3 % after it, while the absolute number of leads remained approximately comparable across the two periods. However, this before-and-after comparison should be interpreted only as descriptive evidence. It does not establish that the model caused the increase, since other changes in lead quality, seasonality, operational processes, or external conditions may also have contributed. Any claim that prioritising high-probability leads raises overall attendance would require a controlled comparison, such as an A/B test, which this work did not carry out. The contribution of the model is thus best stated narrowly: it provides a stable, interpretable, and operationally available ranking of booked leads by attendance likelihood, whose practical worth depends entirely on how it is embedded in the follow-up process.

That embedding is the subject of the remainder of this chapter. Establishing that the score is stable, interpretable, and legitimately computable at prediction time is a precondition for deployment, but it is not deployment itself. For the score to affect outcomes it must be produced repeatedly and reliably against live data, exposed in a form that non-technical teams can act on, and connected to concrete follow-up decisions in the trial-class workflow. The following sections describe how this was done: first the scoring pipeline and its supporting architecture, and then the integration of the resulting buckets into the operational handling of leads.

Figure 8.1 summarizes how the model output is integrated into the operational workflow.

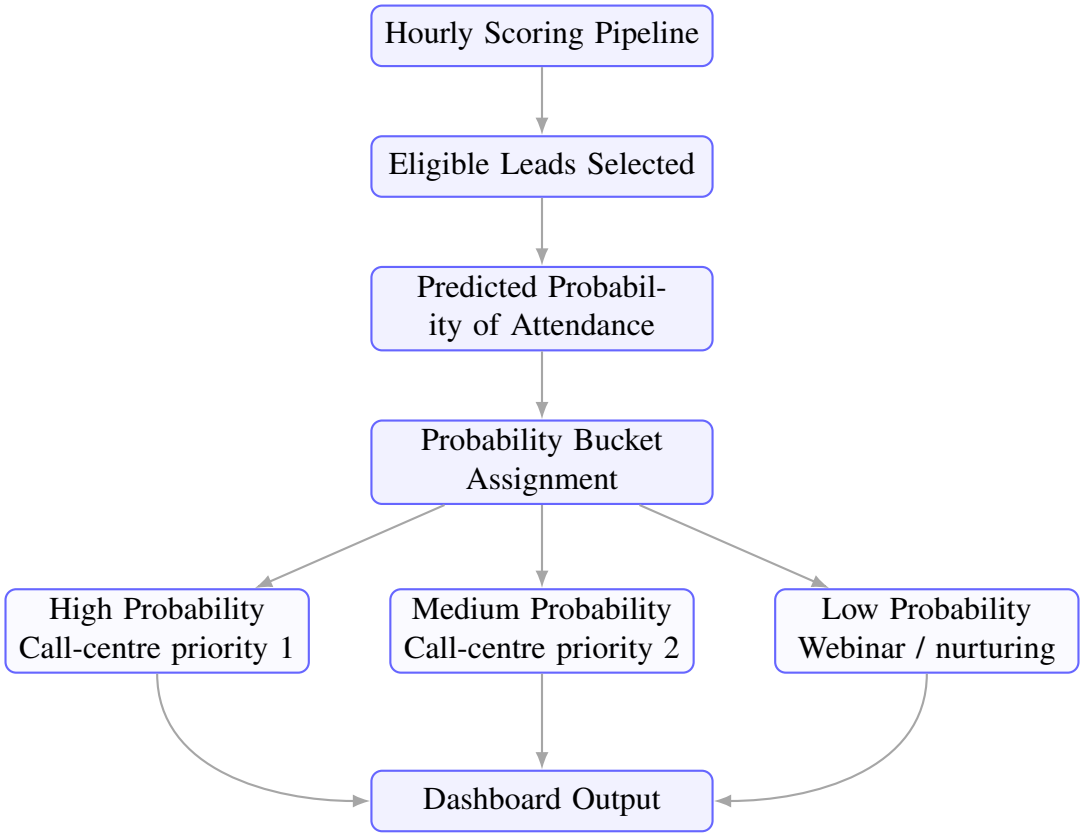

Figure 8.1: Operational integration of the trial-class scoring model

8.2 Deployment Architecture

Because predictive performance was effectively tied across the strongest candidates (Table 7.1), LightGBM was selected for deployment on practical rather than accuracy grounds: its native handling of missing values and mixed-type tabular features, its fast batch inference, and its readily available feature-importance output for monitoring. The next step was to move from a notebook-based experiment to a repeatable scoring process that could be integrated into the existing acquisition workflow.

The deployment was designed as an hourly batch-scoring pipeline rather than a real-time prediction endpoint. This suited the operational setting, since lead prioritisation did not require a millisecond-level response; it was sufficient for newly booked leads to be scored at the next scheduled run and made available to downstream systems within a short time window. Because the qualification quiz and the booking step occur back-to-back in the same session, a lead is eligible for scoring once the booking record is complete, and is scored at the first hourly run thereafter. Consistent with the target

definition, the model then estimates the probability that the booked lead attends the trial class. Since scoring takes place after booking, the scheduled trial-class time already exists in the record and the interval to the class can be used as an input; the constraint that remains is that no feature may draw on information arising after the scoring moment - between the hourly scoring run and the class itself.

At each run, the script selected newly eligible leads who had completed the qualification quiz since the previous execution, applying the same eligibility logic used during model development. In particular, it checked whether the quiz corresponded to the main qualification flow and whether the number of recorded questions was consistent with a complete submission. This check mattered because a record could appear close to the hourly extraction boundary while still incomplete: a submission with only a few recorded questions may indicate that the user had not yet finished the expected flow and should not yet be scored. To handle such cases, the pipeline included a boundary-control mechanism. If a lead had already been scored in a previous run, the script checked whether the current record was more complete or had been updated and, if so, recalculated and reassigned the score. Recalculation was triggered only by greater completeness of the input record, never by information observed after the prediction moment, so this mechanism did not introduce leakage. It made the process robust to delayed data arrival and partial submissions occurring shortly after a run.

For each eligible observation, the script reconstructed the feature vector using the same preprocessing logic as in the modelling stage. Maintaining consistency between the training and production pipelines was necessary to ensure that the model received features in the same format as during development. The trained LightGBM model then generated a predicted probability of attendance, which was converted into an operational bucket using thresholds defined on the training data, see Section 6.9.1. Deriving the low-, medium-, and high-probability boundaries from the training score distribution, rather than from live production data, kept the bucket definitions stable and independent of future observations.

The scoring result was written to a dedicated output table storing the lead or CRM reference, the quiz or booking reference, the scoring timestamp, the predicted probability, the assigned bucket, and technical metadata about the run. Keeping model outputs in a separate table created a stable interface between the machine learning pipeline and downstream reporting or operational systems, without modifying the original source tables. This table was connected to a dashboard (Figure 8.2), allowing business users to monitor the number of newly scored leads, the distribution of leads across buckets, and the evolution of predicted attendance probability over time. The dashboard made the model output accessible to non-technical users without requiring direct interaction with the notebook, scoring script, or model object.

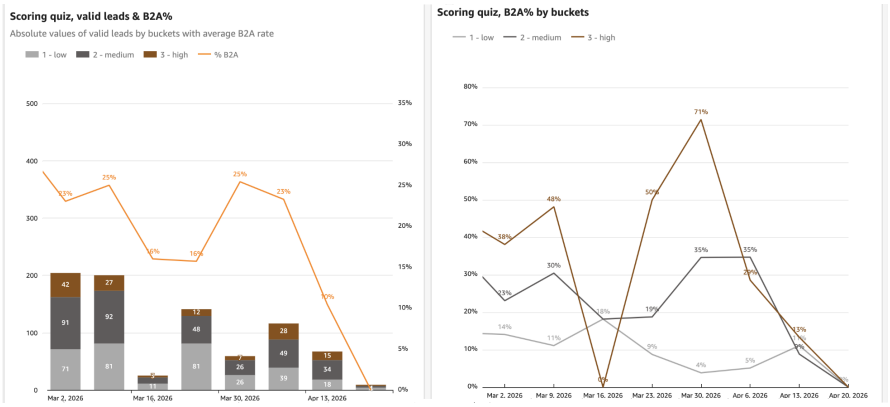

Figure 8.2: Scoring Quiz: Results on a Dashboard

The production setup was implemented in collaboration with a data engineer, since deployment required more than training a model: the scoring script had to be connected to warehouse tables, scheduled for repeated execution, tested on boundary cases, and integrated with the dashboard and call-centre systems. This made the model runnable on a schedule against live warehouse tables rather than only inside a notebook, connecting it to the acquisition funnel as a repeatable scoring process for lead prioritisation.

8.3 Integration with the Trial-Classes Workflow

The scoring output was intended to support operational prioritisation within the trial-class workflow. After each hourly scoring run, leads were assigned to probability buckets, which were then used to differentiate follow-up actions across three tiers of decreasing priority: high, medium, and low.

Leads in the high-probability bucket were passed to the call-centre workflow with the highest priority, so that the team could focus first on those with the greatest predicted likelihood of attending. Medium- probability leads were handled next, immediately after the high-probability group: they entered the same call-centre follow-up process but at a lower priority, ensuring they were still contacted directly once the most promising leads had been addressed. In this respect the model acted as a decision-support layer: it did not replace human judgement or existing business rules, but ordered leads according to expected attendance probability so that limited call-centre capacity could be allocated to the leads most likely to convert.

A separate process was designed for leads in the low-probability bucket. Rather than being prioritised for direct call-centre follow-up, these leads were additionally invited to informational webinars before the trial class. During these webinars, the company introduced the school, explained the learning process, and showed examples of

students' work. The purpose of this intervention was to increase engagement and reduce uncertainty among leads who were less likely to attend according to the model score.

The model output was therefore not used only to identify the most promising leads, but to support differentiated treatment across all three groups: high- and medium-probability leads received direct call-centre attention in order of priority, while low-probability leads were routed to a lower-intensity nurturing process intended to improve motivation and familiarity with the school before the scheduled trial class.

8.4 Limitations

The findings of this thesis must be read against a number of limitations, which fall into two broad groups: those arising from the data available during the internship, and those affecting how far the results can be generalised beyond the specific setting studied.

8.4.1 Data Limitations

The most fundamental limitation is the size and scope of the dataset. The modelling population was restricted to completed qualification quizzes in the Latin American segment, yielding 5,690 lead–quiz-completion observations of which 23.08% were positive. This is a modest sample for a problem with this many candidate features, and it constrains both the complexity of model that can be supported and the precision of the reported estimates. The held-out test set in particular contained 1,138 observations with 244 attendees, so the bucket-level rates and lift figures, while indicative, rest on relatively small counts in each cell and should be treated as estimates with non-trivial uncertainty rather than precise values.

The data were also drawn from a single, recent observation window (quiz completions between 27 October 2025 and 12 March 2026) and a single market segment during a period of active expansion. Campaign composition, quiz design, and acquisition channels changed over this window, and the time-based split means the test period reflects only the most recent few weeks. The narrowing of the bucket gradient from training to test is consistent with this temporal drift, and there is no guarantee that the relationships learned remain stable as campaigns and markets evolve. The model should therefore be regarded as fitted to a particular period rather than as a permanent description of attendance behaviour.

Several features carried substantial missingness, most notably the advertising-creative metadata and the Google Analytics recency variable (`days_since_last_ga_session_asof`, missing in 87.45% of rows). While missingness was handled explicitly and in some cases is itself informative, the high rate limits how much can be concluded from these fields. The campaign-level metrics are also aggregated context rather than individual behaviour:

leads sharing an advertisement share the same campaign values, so these features describe the acquisition environment, not the specific lead. Finally, some quiz signals are self-reported (declared budget, willingness to pay, internet quality), and self-reported intent is known to diverge from behaviour; the unexplained association between high declared spending and lower predicted attendance may be one symptom of this.

A further limitation concerns the small number of records with inconsistent timing (for example, scheduled trial-class times preceding quiz start, found in 1.88% of rows). Although this share is small, it indicates that the underlying timestamp data are not perfectly clean, which is relevant given how central booking-to-class timing is to the model.

A final data constraint concerned the unavailability of certain real-time signals as model inputs. Real-time Google Analytics data could not be used for scoring in the form that would have been ideal, so behavioural web signals were restricted to events recorded at least one day before quiz completion. This limited the freshness of the digital-trace features available to the model and excluded potentially informative last-session behaviour close to the scoring moment.

8.4.2 Generalizability

The results are specific to one company, one market segment, and one funnel design, and several features of this setting limit external validity. The model was trained on Kodland's LatAm quiz-based funnel, in which a qualification quiz precedes an immediate booking step. Companies without a qualification quiz, or with a different booking flow, would not have the same features available and could not reproduce the model directly. The strong role of the booking-to-class interval, in particular, depends on a funnel in which that interval exists and varies in the way it does here.

The scored population is also not the general lead population but the subset who completed the quiz and booked. The overall test attendance rate of 21.4% is far above the historical 7–8% baseline precisely because the qualification step pre-selects more committed leads. Any attempt to apply the model, or to compare its bucket rates, to an unfiltered lead population would be misleading. The model answers a conditional question: given a booking, will the lead attend, and its outputs are meaningful only for that conditioned population.

More broadly, this is a single applied case study rather than a controlled or multi-site study. The modest and stable ROC-AUC around 0.70 is consistent with results reported in the no-show literature across other domains, which provides some reassurance that the difficulty observed here is not anomalous; but a single internship project cannot establish that the approach generalises across markets, languages, or EdTech business models. The contribution is best understood as a demonstration of feasibility and method in one realistic operational setting, not as a generalisable performance claim.

8.5 Ethical Considerations

Embedding a predictive score into a real acquisition process raises ethical questions that are distinct from its technical performance. Two are particularly relevant here: the handling of personal data, and the fairness of using the score to allocate human attention across leads.

8.5.1 Data Privacy and GDPR Compliance

The data used in this work describe identifiable individuals: prospective customers and, indirectly, their children, and therefore fall within the scope of data-protection regulation. Because the host company operates within the European Union, the processing falls within the scope of the General Data Protection Regulation (GDPR), which sets the standard applied throughout this work even where leads were located in Latin American markets. Web-tracking data were processed on the basis of cookie consent obtained at the point of website interaction, and the CRM and quiz data were provided directly by the lead in the course of booking a trial class.

No directly identifying personal data were used in the modelling: the only person-level demographic attribute included as a feature was age, and the CRM identifier used for record linkage was pseudonymised and excluded from the predictive feature set. This is particularly relevant given that the product is aimed at children: the model did not process personal data describing the child beyond the age value, and no child-identifying information entered the feature set. The underlying dataset is additionally governed by a non-disclosure agreement with the host company, for which reason it is not released, and all results reported here are based on anonymised, aggregated outputs.

Independently of the specific legal regime, the processing involved here is operational rather than automated decision-making with legal or similarly significant effects: the score orders follow-up effort and does not, by itself, grant or deny anyone a service. This distinction matters for the proportionality of the processing, because the consequence of a score for an individual lead is a difference in follow-up intensity rather than exclusion from the product. Data minimisation and purpose limitation nonetheless apply, and the feature set should be reviewed periodically to confirm that each retained variable is genuinely necessary for the scoring purpose.

8.5.2 Algorithmic Fairness in Lead Prioritization

The score is used to prioritise human follow-up, which means that errors and biases in the score translate into unequal attention rather than into outright denial of service. This lowers the stakes relative to high-impact automated decisions, but it does not remove the fairness concern: leads placed in the low bucket receive a lower-intensity nurturing

process and less direct call-centre contact, and if the score systematically under-rates certain groups, those groups receive less attention regardless of their true intent.

This concern is sharpened by the features the model relies on. Country is among the three most influential features, and device access (`has_computer`) and connectivity (`speed_internet`) are also prominent. These are correlated with socio-economic status, so a score that down-weights leads with poorer connectivity or phone-only access may systematically de-prioritise less affluent families. Whether this is acceptable depends on the operational goal: it is defensible as a proxy for the genuine practical ability to attend an online class, but it risks entrenching unequal access if treated as a measure of worth rather than of attendance probability.

It should be stated clearly that no formal fairness audit across country, device, or connectivity subgroups was carried out in this work, and this is a limitation. What can be said is that the score was never used to exclude any lead from service: every booked lead was processed and contacted regardless of their feature values, including leads without computer access or with poor connectivity. The score affected the order and intensity of follow-up, not whether a lead was served at all, and the low-probability bucket was routed to a nurturing process rather than dropped. This design limits the harm a biased score could cause, but it does not substitute for a subgroup performance check, which is recommended as future work.

Chapter 9

Conclusion and Future Work

9.1 Summary of Contributions

This thesis examined whether a machine-learning lead-scoring model can support trial-class attendance prediction in a real EdTech acquisition funnel, using an internship at Kodland as the empirical setting. Its contributions are methodological and applied rather than algorithmic.

First, it formalised the prediction task in a way that takes deployment seriously: a supervised binary classification of whether a booked lead attends within seven days, with an explicit scoring moment and a temporal- validity rule that admits a feature only if its value is available at that moment. Second, it constructed a feature set from heterogeneous real sources: CRM, quiz, web-tracking, and campaign metadata, under that constraint, and demonstrated the leakage discipline this requires, including the audit of the booking-to-class interval feature against the scoring moment. Third, it compared linear, tree, and gradient-boosting models under a time-based split and found their ranking performance effectively tied at a modest level (test ROC-AUC around 0.70), with neither feature selection nor hyperparameter tuning improving on the untuned baseline, a finding that locates the performance ceiling in the data rather than the algorithm. Fourth, it turned the continuous score into an interpretable three-bucket segmentation whose attendance gradient is monotonic and material out of sample (8.1%, 19.0%, and 36.5% in the low, medium, and high buckets), and validated this gradient on held-out data. Finally, it embedded the score in an actual operational process, an hourly batch-scoring pipeline feeding a dashboard and differentiated call-centre and nurturing workflow,and reflected on this as a concrete instance of operational digital transformation.

9.2 Answering the Research Questions

The Introduction framed the work around four questions: whether a predictive machine-learning model can improve the efficiency of lead qualification in a real acquisition funnel; what theoretical and methodological basis its practical application requires; how it can be deployed and integrated into existing business processes; and which managerial actions follow.

On the first, the model improves the efficiency of lead qualification not by raising predictive accuracy beyond simple baselines but by reordering finite follow-up effort: it concentrates roughly half of all eventual attendees into the top-scoring third of booked leads, allowing call-centre capacity to be directed where attendance is most likely. On the second, the work shows that the decisive requirements are a correctly defined target and scoring moment, a feature set restricted to information available at that moment, imbalance-aware evaluation rather than accuracy, time-respecting validation, and interpretability used as a leakage audit. On the third, the score was operationalised as a scheduled hourly batch process producing interpretable buckets, surfaced through a dashboard, and connected to tiered follow-up without requiring bespoke infrastructure. On the fourth, the managerial actions that follow are the prioritisation of high- and medium-bucket leads for direct contact and the routing of low-bucket leads to lower-intensity nurturing, with human judgement retained throughout.

9.3 Practical Recommendations for Kodland

Several recommendations follow directly from the findings. First, the model should be used as a triage aid, not as an individual-level predictor: its value is in ordering leads into buckets, and decisions about individual leads should retain human judgement, given the modest discrimination. Second, because the performance ceiling is set by the data rather than the algorithm, future effort should be directed at better features and cleaner data, for example richer behavioural signals available at scoring time, rather than at further model tuning, which this work showed yields no gain. Third, the unexplained high-spend association and the reliance on socio-economically correlated features (country, device, connectivity) should be reviewed with the business before the score is trusted for those segments. Fourth, because the model is fitted to one period and market, it should be monitored and periodically refitted, with input and outcome distributions watched for drift.

Fifth, before the score is credited with improving attendance, Kodland should run a controlled experiment in which comparable booked leads are randomly assigned to score-driven prioritisation or to the existing process. This is the only way to establish that acting on the score causally raises attendance rather than merely ranking leads who would have attended anyway, and it would convert the projected efficiency gains reported

in this thesis into a measured business effect.

9.4 Directions for Future Research

Four directions follow from the limitations. First, and most important, a controlled experiment is needed to establish whether prioritising high-probability leads causally increases attendance; this thesis quantifies how attendees are distributed across buckets but cannot show that acting on the buckets changes outcomes. Second, the approach should be tested across additional markets, languages, and funnel designs to assess generalisability beyond the single LatAm segment studied here. Third, a formal fairness audit across country, device, and connectivity groups would clarify whether the score systematically de-prioritises less affluent families, and whether mitigation is warranted. Fourth, richer scoring-time features, particularly behavioural signals not dependent on self-report, may be the only route to raising the performance ceiling that data, not algorithm choice, currently imposes. Probability calibration, only briefly addressed here, would also be worth developing if the score's magnitudes (not just its ordering) are ever used directly.

Bibliography

[1] Kjersti Aas, Martin Jullum, and Anders Løland. Explaining individual predictions when features are dependent: More accurate approximations to shapley values. *Artificial Intelligence*, 298:103502, 2021.

[2] Alexei Alexandrov and Martin A. Lariviere. Are reservations recommended? *Manufacturing & Service Operations Management*, 14(2):218–230, 2012.

[3] Neil T. Bendle, Paul W. Farris, Phillip E. Pfeifer, and David J. Reibstein. *Marketing Metrics: The Manager's Guide to Measuring Marketing Performance*. Pearson, 4 edition, 2020.

[4] Christoph Bergmeir and José M. Benítez. On the use of cross-validation for time series predictor evaluation. *Information Sciences*, 191:192–213, 2012.

[5] James Bergstra and Yoshua Bengio. Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13:281–305, 2012.

[6] Robert C. Blattberg, Byung-Do Kim, and Scott A. Neslin. *Database Marketing: Analyzing and Managing Customers*. Springer, New York, NY, 2008.

[7] Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[8] Glenn W. Brier. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78(1):1–3, 1950.

[9] Carrie J. Cai, Samantha Winter, David Steiner, Lauren Wilcox, and Michael Terry. "Hello AI": Uncovering the onboarding needs of medical practitioners for human-AI collaborative decision-making. In *Proceedings of the ACM on Human-Computer Interaction (CSCW)*, volume 3, pages 104:1–104:24, 2019.

[10] Danaë Carreras-García, David Delgado-Gómez, Fernando Llorente-Fernández, and Ana Arribas-Gil. Patient no-show prediction: A systematic literature review. *Entropy*, 22(6):675, 2020.

[11] Tianqi Chen and Carlos Guestrin. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794, 2016.

[12] David Court, Dave Elzinga, Susan Mulder, and Ole Jørgen Vetvik. The consumer decision journey. *McKinsey Quarterly*, 3:96–107, 2009.

[13] Hannes Datta, Bram Foubert, and Harald J. van Heerde. The challenge of retaining customers acquired with free trials. *Journal of Marketing Research*, 52(2):217–234, 2015.

[14] Thomas H. Davenport and Rajeev Ronanki. Artificial intelligence for the real world. *Harvard Business Review*, 96(1):108–116, 2018.

[15] Anna Veronika Dorogush, Vasily Ershov, and Andrey Gulin. Catboost: gradient boosting with categorical features support. In *Proceedings of the Workshop on Machine Learning Systems at NeurIPS*, 2018.

[16] Tom Fawcett. An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8):861–874, 2006.

[17] Mirta Galesic and Michael Bosnjak. Effects of questionnaire length on participation and indicators of response quality in a web survey. *Public Opinion Quarterly*, 73(2):349–360, 2009.

[18] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Series in Statistics. Springer, New York, NY, 2 edition, 2009.

[19] Haibo He and Edwardo A. Garcia. Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*, 21(9):1263–1284, 2009.

[20] Haibo He and Edwardo A. Garcia. Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*, 21(9):1263–1284, 2009.

[21] David W. Hosmer, Stanley Lemeshow, and Rodney X. Sturdivant. *Applied Logistic Regression*. Wiley, Hoboken, NJ, 3rd edition, 2013.

[22] Itir Karaesmen and Garrett van Ryzin. Overbooking with substitutable inventory classes. *Operations Research*, 52(1):83–104, 2004.

[23] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. LightGBM: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems 30 (NeurIPS)*, pages 3146–3154, 2017.

[24] Kodland. Internal database, 2026. Internal company database used for thesis analysis, accessed March 2026.

[25] Philip Kotler, Hermawan Kartajaya, and Iwan Setiawan. *Marketing 5.0: Technology for Humanity*. Wiley, Hoboken, NJ, 2021.

[26] Peter S. H. Leeflang, Peter C. Verhoef, Peter Dahlström, and Tjark Freundt. Challenges and solutions for marketing in a digital era. *European Management Journal*, 32(1):1–12, 2014.

[27] David Loshin. *The Practitioner's Guide to Data Quality Improvement*. Morgan Kaufmann, Burlington, MA, 2010.

[28] Scott M. Lundberg, Gabriel Erion, Hugh Chen, Alex DeGrave, Jordan M. Prutkin, Bala Nair, Ronit Katz, Jonathan Himmelfarb, Nisha Bansal, and Su-In Lee. From local explanations to global understanding with explainable AI for trees. *Nature Machine Intelligence*, 2(1):56–67, 2020.

[29] Scott M. Lundberg, Gabriel Erion, Hugh Chen, Alex DeGrave, Jordan M. Prutkin, Bala Nair, Ronit Katz, Jonathan Himmelfarb, Nisha Bansal, and Su-In Lee. From local explanations to global understanding with explainable AI for trees. *Nature Machine Intelligence*, 2(1):56–67, 2020.

[30] Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems 30 (NIPS 2017)*, pages 4765–4774, 2017.

[31] Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems 30*, pages 4765–4774, 2017.

[32] Robert Monarch. *Human-in-the-Loop Machine Learning: Active Learning and Annotation for Human-Centered AI*. Manning Publications, Shelter Island, NY, 2021.

[33] Jamie P. Monat. Industrial sales lead conversion modeling. *Marketing Intelligence & Planning*, 29(2):178–194, 2011.

[34] Scott A. Neslin, Dhruv Grewal, Robert Leghorn, Venkatesh Shankar, Marije L. Teerling, Jacquelyn S. Thomas, and Peter C. Verhoef. Challenges and opportunities in multichannel customer management. *Journal of Service Research*, 9(2):95–112, 2006.

[35] Alexandru Niculescu-Mizil and Rich Caruana. Predicting good probabilities with supervised learning. In *Proceedings of the 22nd International Conference on Machine Learning*, pages 625–632. Association for Computing Machinery, 2005.

[36] John B. Norris, Chetan Kumar, Suresh Chand, Herbert Moskowitz, Susan A. Shade, and Deanna R. Willis. An empirical investigation into factors affecting patient cancellations and no-shows at outpatient clinics. *Decision Support Systems*, 57:428–443, 2014.

[37] Liudmila Prokhorenkova, Gleb Gusev, Aleksandr Vorobev, Anna Veronika Dorogush, and Andrey Gulin. Catboost: unbiased boosting with categorical features. *Advances in Neural Information Processing Systems*, 31, 2018.

[38] Foster Provost and Tom Fawcett. *Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking*. O'Reilly Media, Sebastopol, CA, 2013.

[39] Katharina Reinecke, Minh Khoa Nguyen, Abraham Bernstein, Michael Näf, and Krzysztof Z. Gajos. Doodle around the world: Online scheduling behavior reflects cultural differences in time perception and group decision-making. In *Proceedings of the 17th ACM Conference on Computer Supported Cooperative Work and Social Computing (CSCW)*, pages 45–54, 2014.

[40] Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5):206–215, 2019.

[41] Takaya Saito and Marc Rehmsmeier. The precision–recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. *PLOS ONE*, 10(3):e0118432, 2015.

[42] D. Sculley, Gary Holt, Daniel Golovin, Eugene Davydov, Todd Phillips, Dietmar Ebner, Vinay Chaudhary, Michael Young, Jean-François Crespo, and Dan Dennison. Hidden technical debt in machine learning systems. In *Advances in Neural Information Processing Systems 28 (NIPS 2015)*, pages 2503–2511, 2015.

[43] Neil Selwyn. *Education and Technology: Key Issues and Debates*. Bloomsbury Academic, London, 2nd edition, 2016.

[44] Galit Shmueli and Otto R. Koppius. Predictive analytics in information systems research. *MIS Quarterly*, 35(3):553–572, 2011.

[45] Gregory Vial. Understanding digital transformation: A review and a research agenda. *The Journal of Strategic Information Systems*, 28(2):118–144, 2019.

[46] Gregory Vial. Understanding digital transformation: A review and a research agenda. *The Journal of Strategic Information Systems*, 28(2):118–144, 2019.

[47] Ben Williamson and Anna Hogan. Commercialisation and privatisation in/of education in the context of Covid-19. Technical report, Education International, Brussels, Belgium, 2020.

[48] Olaf Zawacki-Richter, Victoria I. Marín, Melissa Bond, and Franziska Gouverneur. Systematic review of research on artificial intelligence applications in higher education – where are the educators? *International Journal of Educational Technology in Higher Education*, 16(39), 2019.