Conformal Predictive Distributions for Order Fulfillment Time Forecasting

Tinghan Ye, Amira Hijazi, and Pascal Van Hentenryck

H. Milton Stewart School of Industrial & Systems Engineering, Georgia Institute of Technology, USA
{joe.ye,ahijazi6,pvh}@gatech.edu

arXiv:2505.17340v2 [cs.LG] 1 Aug 2025

Abstract. Accurate estimation of order fulfillment time is critical for e-commerce logistics, yet traditional rule-based approaches often fail to capture the inherent uncertainties in delivery operations. This paper introduces a novel framework for distributional forecasting of order fulfillment time, leveraging Conformal Predictive Systems and Cross Venn-Abers Predictors—model-agnostic techniques that provide rigorous coverage or validity guarantees. The proposed machine learning methods integrate granular spatiotemporal features, capturing fulfillment location and carrier performance dynamics to enhance predictive accuracy. Additionally, a cost-sensitive decision rule is developed to convert probabilistic forecasts into reliable point predictions. Experimental evaluation on a large-scale industrial dataset demonstrates that the proposed methods generate competitive distributional forecasts, while machine learning-based point predictions significantly outperform the existing rule-based system—achieving up to 14% higher prediction accuracy and up to 75% improvement in identifying late deliveries.

Keywords: distributional forecasting, conformal prediction, order fulfillment, logistics

1 Introduction

In the e-commerce industry, rule-based approaches are widely used to estimate delivery dates for ship-to-home order fulfillment. These approaches typically rely on static transit tables that specify carrier transit times, scheduled pickup windows, and valid days for pickup, transit, and delivery. While straightforward to implement, such static configurations often fall short in capturing the dynamic and stochastic nature of real-world logistics operations, particularly as supply chain conditions evolve over time [9].

Recently, data-driven approaches–particularly machine learning (ML) methods—have gained traction for estimating delivery times, particularly among large e-commerce companies such as Amazon and JD.com [20, 24]. Despite this increasing adoption, building accurate predictive models for order fulfillment time remains inherently difficult due to the complexity and variability of the fulfillment life cycle. As illustrated in Fig. 1, the process begins with the customer offer stage. Once an order is placed, it enters the location selection phase, during which a Fulfillment Location (FL)—such as a warehouse or a retail store—is selected based on inventory availability, customer proximity, and operational constraints. At this stage, a tentative carrier is identified to estimate shipping costs and delivery times, although the final carrier may be confirmed later in the process. The requested stock-keeping unit (SKU) is then retrieved, packed, and staged for pickup according to the operational procedures of the selected fulfillment location. The shipment stage begins with final carrier selection and the generation of a shipping label, marking the point at which the designated carrier assumes responsibility for the order. Last-mile delivery methods can vary, but commonly, the parcel is picked up from the fulfillment location, transported to an origin hub, and moved through the carrier's network until it reaches the customer.

There is a growing need in the field to move beyond deterministic point predictions and embrace *distributional forecasting*, which better captures the uncertainty inherent in order fulfillment time, particularly given the involvement of multiple stakeholders such as retailers, warehouses, and carriers. By modeling the full range of possible outcomes, distributional forecasting offers a more faithful representation of uncertainty and serves as a critical input for contextual stochastic optimization models [33].

Beyond logistics, distributional forecasting has gained significant attention in domains such as energy systems [13, 36, 21] and healthcare [3]. Compared to quantile or interval forecasts, which represent limited aspects of uncertainty, distributional forecasts provide a complete view of the predictive distribution. This allows for the consistent derivation of point estimates, quantiles, and prediction intervals—capabilities that cannot be recovered in reverse from quantile-based approaches alone [30]. By more thoroughly quantifying uncertainty, distributional forecasting empowers businesses to adopt risk-aware operational strategies that strike a balance between service reliability and cost efficiency, grounded in rigorous statistical guarantees.

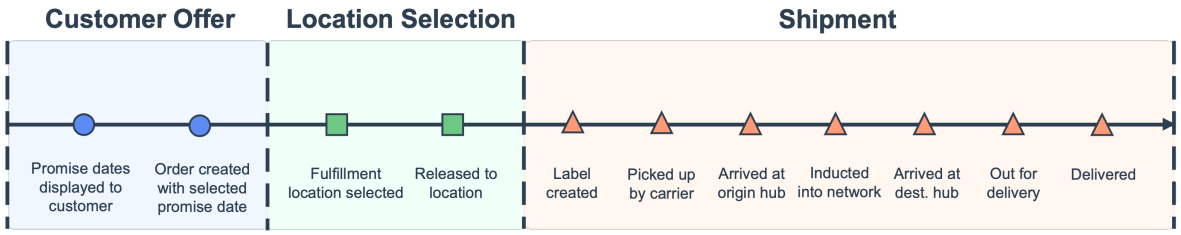


Fig. 1: Example of Order Fulfillment Life Cycle.

1.1 Contributions

This paper introduces a novel framework for the distributional forecasting of order fulfillment time, leveraging conformal prediction, a model-agnostic, distribution-free framework that provides robust coverage/validity guarantees under minimal assumptions. Specifically, Conformal Predictive Systems (CPS) [30] and Cross Venn-Abers Predictors (CVAP) are applied for probability density forecasting. To the best of the authors' knowledge, this is the first application of CPS and CVAP in delivery service and travel time estimation, establishing a new paradigm for generating reliable predictive distributions. The main contributions are summarized as follows:

1. The paper pioneers the use of CPS and CVAP for distributional forecasting in service time estimation, allowing flexible integration with any regression or multi-class classification model, including advanced deep learning architectures.
2. The paper develops a novel two-stage classify-then-regress model that synergistically combines CPS and CVAP to produce more informative probabilistic forecasts.
3. The paper proposes a cost-sensitive decision rule to convert conformal-calibrated distributions into flexible point predictions, balancing predictive accuracy with business cost considerations.
4. The paper introduces granular spatial-temporal features to capture the operational performance of fulfillment locations and carriers, enhancing both predictive power and interpretability.
5. The paper validates the proposed framework on a real-world industrial dataset comprising more than 5 million orders across over one year, demonstrating that the proposed distributional forecasts are competitive with established methods, while the ML-based point predictions substantially outperform the existing rule-based system.

The methodologies proposed in this paper are already being developed and tested by the industrial partner alongside their existing infrastructure. In the initial rollout, they will augment the current fulfillment decision pipeline by providing probabilistic delivery-time guidance in tandem with rule-based estimates.

1.2 Related Work

Delivery service time forecasting is a critical aspect of logistics operations. A recent survey on this topic is provided in [31]. The specific focus of this paper is parcel delivery service operations in the context of online retail logistics. Prior studies have addressed similar challenges, including [32, 2, 15, 24, 25, 38, 7, 34, 37, 33]. While most papers focus on estimating delivery dates at the customer offer stage to set effective customer promise dates, this paper targets predictions at the time of location selection—that is, immediately after order creation but before assignment to a fulfillment location or carrier. This distinction aligns the study more closely with [33], where estimated delivery dates are leveraged for downstream contextual stochastic optimization to determine a fulfillment location-carrier pair. This work extends beyond prior research by [33] in three significant ways: it introduces novel methodologies for distributional forecasting, incorporates more granular spatio-temporal features particularly related to carrier operations, and validates these advances on a more comprehensive and longitudinal dataset.

Methodologically, this work is closely related to research on probabilistic forecasting for service or travel time estimation. Existing approaches can be generally categorized into parametric and non-parametric methods. Parametric methods assume that uncertainty follows a specific probability distribution [26, 16, 23] (e.g., a log-normal distribution as in [16]), whereas non-parametric methods make no such assumptions and often employ ML–based approaches. Common ML techniques include tree-based ensemble methods [12, 25, 3, 33] and deep neural networks [16, 11].

These models are typically framed as either Regression or Multi-Class Classification (MC-CLF) problems. Regression-based approaches often rely on specialized model structures, such as Quantile Random Forest (QRF) [19], to generate predictive distributions [25], while classification-based methods naturally produce discrete probability distributions [16, 33].

Despite its potential, conformal prediction [28, 18, 14] has not been widely explored for probabilistic parcel delivery time forecasting. A concurrent study [11] applies conformal prediction within a classify-then-regress framework and utilizes conformalized quantile regression to construct prediction intervals on a related problem. Other data-driven methods have addressed related operational problems—such as inbound load plan adjustments—by applying two-stage conformal prediction to produce confidence-aware outputs [6]. However, these methods are limited to prediction set or interval forecasting. In contrast, the framework proposed in this paper explicitly focuses on probability density forecasting, which provides richer information and greater utility for decision-making applications.

This paper also contributes to the literature on cost-centric decision rules, building upon methods previously established by [25, 7] in the context of delivery time estimation.

2 Problem Definition

This section revisits the problem of order fulfillment time distributional forecasting, first introduced in [33]. The goal is to estimate the probabilistic distribution of fulfillment time deviation from the customer's promised date, leveraging relevant contextual information. Given a predicted deviation, one can readily compute the estimated delivery date by adding the deviation to the promised date.

Each instance i represents an incoming customer order along with a candidate fulfillment location and an eligible carrier to fulfill and ship the order. Let $\mathcal{X}$ be the covariate (feature) space and $x_i \in \mathcal{X}$ be the covariates associated with instance i. Each covariate includes features observable from the order, the fulfillment location, and the carrier.

Let $\mathcal{Y} = \{c_1, c_2, \ldots, c_K\} \subset \mathbb{Z}$ represent the set of possible fulfillment time deviations, measured in days. The elements of $\mathcal{Y}$ are ordered such that $c_1 < c_2 < \ldots < c_K$, and reflect discrete deviation values relative to the promised delivery date. For each order instance i, let Y_i be a random variable that captures the deviation from the promised delivery date, with its realized value denoted by $y_i \in \mathcal{Y}$. Negative values of y_i correspond to early deliveries, zero indicates on-time delivery, and positive values indicate late deliveries.

Given a set of unseen instances $\mathcal{I}$ and their associated covariates, the goal of a forecasting model is to predict the target fulfillment time deviation y_i by producing an estimate $\hat{y}_i$ for each instance $i \in \mathcal{I}$. While traditional approaches focus on point estimation, *distributional forecasting* aims at predicting the full conditional cumulative distribution function (CDF) of the target variable, defined as $F(y_i \mid x_i) = \mathbb{P}(Y_i \leq y_i \mid x_i)$. The objective is to learn an estimated function $\hat{F}(y_i \mid x_i) = \hat{\mathbb{P}}(Y_i \leq y_i \mid x_i)$, that closely approximates the true conditional CDF. By modeling the entire distribution of possible fulfillment time deviations, distributional forecasting enables more informative and risk-aware decision-making compared to single-point predictions.

Let Θ denote the set of learnable model parameters, and let $\hat{F}(x_i; \Theta)$ represent the predicted CDF for instance i, parameterized by Θ. Let $\mathcal{L}$ be a suitable loss function that measures the discrepancy between the predicted distribution $\hat{F}(x_i; \Theta)$ and the observed outcome y_i. Let $\mathcal{D}_{\text{tr}}$ be the set of training instances. The learning task can be formulated as an empirical risk minimization problem:

$$\min_{\Theta} \frac{1}{|\mathcal{D}_{\text{tr}}|} \sum_{i \in \mathcal{D}_{\text{tr}}} \mathcal{L}\left(\hat{F}(x_i; \Theta), y_i\right).$$

One key advantage of distributional forecasting is that it enables efficient and flexible uncertainty quantification without requiring model retraining or recalibration. Specifically, any desired quantile of the predicted fulfillment time deviation can be extracted directly from the predicted CDF, supporting a wide range of operational decisions. The q-th quantile ($q \in [0, 100]$) is defined as: $\hat{y}_i^q = \inf\left\{y : \hat{\mathbb{P}}(Y_i \leq y \mid x_i) \geq q\right\}$.

This formulation allows practitioners to tailor predictions based on risk preferences and service-level objectives. For instance, a lower quantile (e.g., $q = 10$) can be used to generate *conservative estimates* of delivery time, useful for setting aggressive internal targets or evaluating best-case scenarios. The median ($q = 50$) provides a *typical expected deviation*, often used for customer-facing delivery promises. An upper quantile (e.g., $q = 90$) offers a *pessimistic but cautious estimate*, helpful for planning buffer stock or adjusting carrier dispatch windows.

Moreover, prediction intervals can be constructed by selecting appropriate quantile pairs. For example, a 95% prediction interval for the fulfillment time deviation is given by: $[\hat{y}_i^{2.5}, \hat{y}_i^{97.5}]$, which captures the range within which

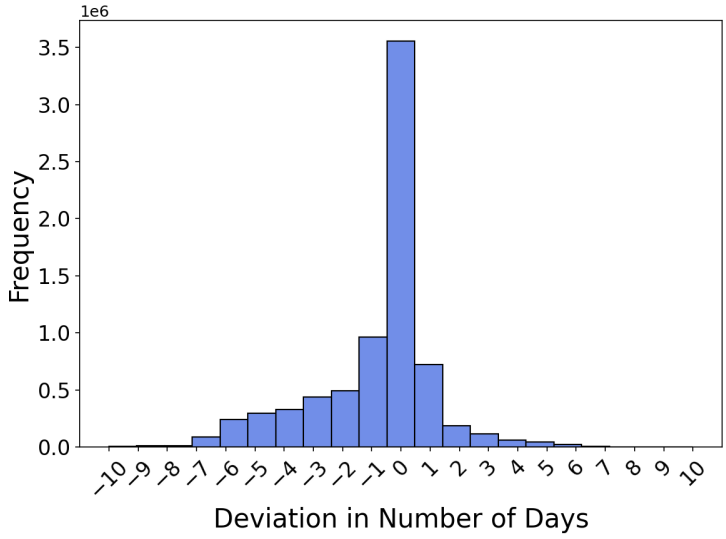


Fig. 2: Distribution of Fulfillment Time Deviations for Orders Shipped via the Selected Carrier.

the deviation is expected to fall with high confidence. In a logistics context, such intervals are valuable for *capacity planning*, *carrier selection*, and *customer communication*, as they provide a data-driven way to quantify delivery risk under uncertainty.

3 Data Description

This study utilizes an anonymized e-commerce transactional order dataset provided by an industrial partner. The dataset contains over 5 million representative ship-to-home orders collected between July 2023 and November 2024. For demonstration purposes, the analysis focuses on orders shipped via a heavily utilized ground shipping services recommended by the industrial partner. However, the proposed framework is designed to be readily extensible to other carriers and service types. To mitigate the influence of extreme outliers, orders with fulfillment time deviations beyond the range of $[-10, 10]$ days—that is, delivered more than 10 days earlier or later than expected—were excluded from the dataset.

The dataset's features can be categorized into three groups: order-level, fulfillment location-level, and carrier-level features. In addition to the features outlined in [33], this study incorporates additional data for each category, including customer membership tier and various timestamps within the fulfillment location and carrier network, making the dataset significantly more comprehensive. Following feature engineering (detailed in Section 4.1), all features utilized in the ML models were summarized in Table 4 in the Appendix.

Fig. 2 illustrates the empirical distribution of fulfillment time deviations for orders shipped using the carrier examined in this study. The distribution is centered around zero, indicating that a substantial proportion of orders are delivered on time. However, a noticeable spread exists on both sides, reflecting instances of early and late deliveries. In logistics operations, any deviation—whether ahead of or behind the promised delivery date—can disrupt planning, reduce customer satisfaction, and introduce inefficiencies into downstream processes [25, 10].

4 Methodology

This section introduces the proposed forecasting framework, illustrated in Fig. 3. The framework first constructs dynamic spatio-temporal representations from raw features, which serve as inputs to either a deterministic regression model or a multi-class classification (MC-CLF) model. To quantify uncertainty, conformal prediction techniques are applied to calibrate the underlying models, generating distributional forecasts—with Conformal Predictive System (CPS) used for regression models and Cross Venn-Abers Predictors (CVAP) for MC-CLF models. Additionally, the framework incorporates an optional cost-sensitive decision rule, which converts conformal calibrated probability distributions into point predictions, ensuring alignment with operational objectives.

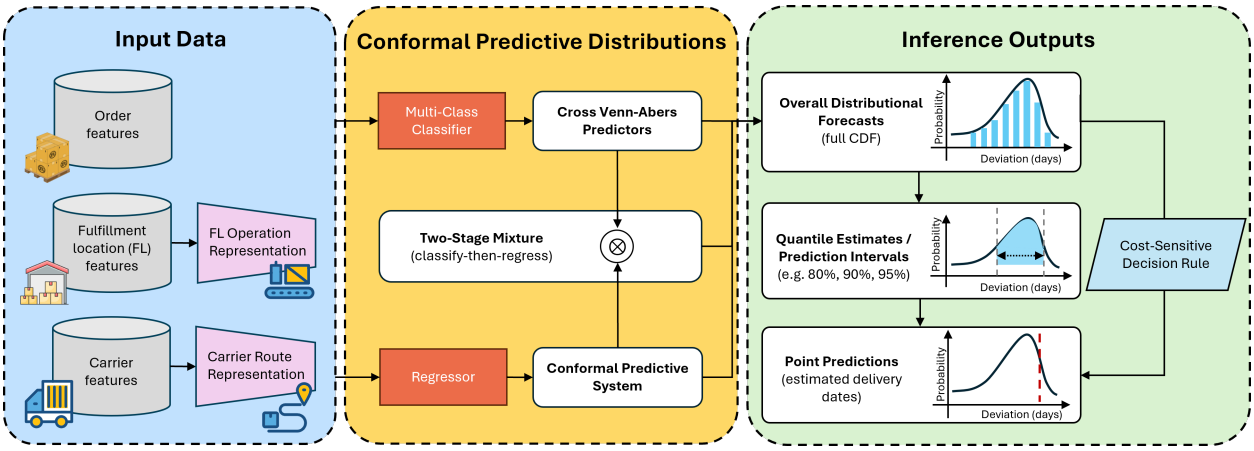


Fig. 3: An Overview of the Conformal Predictive Distribution Framework for Order Fulfillment Time Forecasting.

4.1 Dynamic Spatio-Temporal Representation

The first step of the implementation transforms raw inputs into representative features for ML models. These features capture information about the fulfillment locations and the carrier routes.

Fulfillment Location Operation Representation To represent real-time congestion in each location, the ML models include a feature that captures the waiting time at the time of order placement. To quantify workload intensity, the shipping volume in the past two hours is extracted as a proxy for recent operational throughput. For seasonal workload patterns, the ML models leverage the domain expertise from industrial practitioners to segment each FL's operation time based on wave schedules, operational hours, and facility type. Within each segment, summary statistics of historical processing times were computed using training data.

Carrier Route Representation Since the carrier plays a crucial role in the shipment stage of order fulfillment, the proposed framework integrates a spatio-temporal representation of carrier performance, derived from anonymized historical carrier tracking data. This representation captures key routing characteristics by aggregating information on shipment patterns, carrier hub activity, and transit times. For each origin-destination (O-D) pair, defined at the ZIP5-to-ZIP5 level, the framework extracts aggregated metrics to estimate typical carrier routes. Shipment metrics include the total number of shipments, median number of intermediate stops, and median number of tracking updates per shipment. Exception metrics quantify disruptions, such as the number of shipments delayed by the carrier, weather, or other issues.

To model hub dynamics, the most frequently used origin and destination hubs are identified for each O-D pair. Their latitude and longitude are recorded, and the Haversine distance between them is calculated to represent the common spatial path. Temporal characteristics are captured through the average and standard deviation of key time intervals, including the time from label creation to the initial scan at the origin hub, the induction time into the carrier network, the transit time from the origin scan to the destination scan, and the times from destination scan to "out for delivery" and from "out for delivery" to final delivery. For unseen O-D pairs in the test set, the framework uses a hierarchical imputation strategy, aggregating geographic units from ZIP5 to ZIP3, and then to ZIP2 levels, ensuring robust route estimation based on the most granular available data.

4.2 Conformal Predictive Distribution for Regression and Classification

Conformal Predictive System The *Conformal Predictive System for Regression* (CPS) is a flexible, model-agnostic framework for constructing predictive distributions that offer formal uncertainty quantification. Unlike traditional approaches that rely on strong distributional assumptions or asymptotic guarantees, CPS provides finite-sample, distribution-free coverage under the mild assumption of exchangeability(i.e., the order of data points doesn't affect

the underlying probability distribution) [30]. This makes it particularly well-suited for real-world applications—such as logistics—where the data-generating process is often complex, non-stationary, or only partially understood.

In logistics operations, decisions are frequently made under uncertainty, whether one is forecasting fulfillment times, estimating shipment delays, or allocating resources across stochastic demand. CPS enables such decisions to be informed, not only by point predictions, but also by valid, data-driven confidence estimates that reflect the model's empirical error behavior. By transforming regression outputs into full predictive distributions, CPS supports robust risk assessment and reliability-aware planning.

The following paragraphs describe two implementations of this framework: the *Split Conformal Predictive System (SCPS)*, which offers globally calibrated uncertainty estimates, and the *Mondrian Conformal Predictive System (MCPS)*, which introduces local adaptation to improve calibration in heterogeneous prediction settings.

Split Conformal Predictive System The SCPS is a model-agnostic framework that transforms any regression model's point estimate into a full predictive distribution with finite-sample coverage guarantees. This is particularly useful in logistics settings, where outcome variability—such as delays or fulfillment deviations—can arise from unpredictable operational conditions.

1. **Data Splitting:** Partition the training dataset $\mathcal{D}_{\text{tr}} = \{(x_i, y_i)\}_{i=1}^{N}$ into two disjoint subsets: a *proper training set* $\mathcal{D}_t$, used to train the regression model; a *calibration set* $\mathcal{D}_c = \{(x_j, y_j)\}_{j=1}^{N_c}$, used to quantify prediction uncertainty. This separation allows the model to be evaluated on unseen data when constructing its prediction intervals, improving generalization.
2. **Model Training:** Train a regression model $h : \mathcal{X} \to \mathbb{R}$ using the proper training set $\mathcal{D}_t$. The output $h(x)$ is a point prediction for a new input x, i.e., the expected fulfillment time deviation.
3. **Calibration Score Computation:** For a new input x and for each calibration example $(x_j, y_j) \in \mathcal{D}_c$, compute the calibration score $C_j = h(x) + (y_j - h(x_j))$, where $y_j - h(x_j)$ represents the prediction error for a historical instance. By adding this residual to the prediction for the test instance $h(x)$, the method generates a set of plausible outcomes that reflect how the model has erred in the past. This technique aligns with practical logistics intuition: it adjusts today's prediction based on past deviations, under the assumption that similar mistakes might reoccur.
4. **Predictive Distribution Construction:** Sort the calibration scores to obtain an ordered set: $C_{(1)} < C_{(2)} < \cdots < C_{(N_c)}$, and define boundaries $C_{(0)} = -\infty$, $C_{(N_c+1)} = \infty$. Let $\tau \sim \text{Uniform}(0, 1)$ be a random variable used to smooth the predictive distribution. Then, the estimated CDF can be computed as follows:

$$\hat{F}(y \mid x) = \begin{cases} \frac{n+\tau}{N_c+1} & \text{if } y \in (C_{(n)}, C_{(n+1)}) \text{ for } n \in \{0, \dots N_c\}, \\ \frac{n'-1+(n''-n'+2)\tau}{N_c+1} & \text{if } y = C_{(n)} \text{ for } n \in \{1, \dots N_c\}, \end{cases}$$

where $n' = \min\{m : C_{(m)} = C_{(n)}\}$ and $n'' = \max\{m : C_{(m)} = C_{(n)}\}$.
This function $\hat{F}(y \mid x)$ represents the probability that the true outcome will be less than or equal to y, based on the empirical distribution of historical model errors. As a result, it provides calibrated, data-driven uncertainty estimates. In logistics contexts, this allows for the construction of predictive intervals that quantify risks in terms of fulfillment deviations. For instance, a narrow interval centered around zero indicates high model confidence that a shipment will arrive on time, while a wider interval may suggest increased likelihood of delay—prompting preemptive interventions such as route adjustments or customer notifications.

Mondrian Conformal Predictive System While SCPS provides a globally calibrated predictive distribution, it assumes that the model's prediction errors are uniformly distributed across all inputs. In many real-world settings, and particularly in logistics, this assumption may not hold. For example, the uncertainty associated with predicting short, on-time deliveries may differ markedly from that of predicting extreme delays. The MCPS addresses this by partitioning the calibration data according to the model's own predictions, thereby producing *locally adaptive* predictive distributions [5]. MCPS introduces the following modification to the SCPS framework:

1. **Partitioning by Predicted Output:** Define $B \in \mathbb{N}$ as the number of bins (a tunable hyperparameter). Partition the calibration set $\mathcal{D}_c = \{(x_j, y_j)\}_{j=1}^{N_c}$ into B disjoint subsets $\mathcal{D}_c^1, \dots, \mathcal{D}_c^B$, based on the predicted values $h(x_j)$. Each bin $\kappa \in \{1, \dots, B\}$ corresponds to a range of predicted deviations. For example, one bin may cover near-zero (on-time) predictions, while others may correspond to moderate or extreme delays. This allows MCPS to tailor uncertainty quantification to the model's confidence in different prediction regimes.

2. **Calibration Score Computation (Local):** For a test input x, determine the bin κ to which $h(x)$ belongs. Then compute calibration scores using only the data in the corresponding subset $\mathcal{D}_c^\kappa = \{(x_j, y_j)\}_{j=1}^{N_c^\kappa}$, as follows: $C_j^\kappa = h(x) + (y_j^\kappa - h(x_j^\kappa))$ for $j = \{1, \ldots, N_c^\kappa\}$. As in SCPS, these scores simulate plausible outcomes for the test case by adjusting its prediction using the historical errors observed in similar prediction contexts.
3. **Predictive Distribution Construction (Local):** Sort the bin-specific calibration scores: $C_{(1)}^\kappa < C_{(2)}^\kappa < \cdots < C_{(N_c^\kappa)}^\kappa$, and define boundary values $C_{(0)}^\kappa = -\infty$, $C_{(N_c^\kappa+1)}^\kappa = \infty$. Let $\tau \sim \text{Uniform}(0, 1)$ as before. Then, the bin-specific CDF is:

$$\hat{F}(y \mid x) = \begin{cases} \frac{n+\tau}{N_c^\kappa+1} & \text{if } y \in (C_{(n)}^\kappa, C_{(n+1)}^\kappa) \text{ for } n \in \{0, \ldots N_c^\kappa\}, \\ \frac{n'-1+(n''-n'+2)\tau}{N_c^\kappa+1} & \text{if } y = C_{(n)}^\kappa \text{ for } n \in \{1, \ldots N_c^\kappa\}, \end{cases}$$

where $n' = \min\{m : C_{(m)}^\kappa = C_{(n)}^\kappa\}$ and $n'' = \max\{m : C_{(m)}^\kappa = C_{(n)}^\kappa\}$.
This localized predictive distribution reflects the empirical error behavior for predictions in the same range as the test input. In contrast to SCPS, which pools all calibration scores, MCPS only uses scores from cases where the model made similar predictions. In logistics, this enables more refined uncertainty quantification. For instance, the model may be highly confident in predicting on-time deliveries, resulting in narrower intervals for such cases. Conversely, predictions associated with rare or extreme delays may yield wider intervals, signaling greater uncertainty. This differentiation can support risk-sensitive operational decisions, such as prioritizing high-risk shipments for additional monitoring or contingency planning.

Venn-Abers Calibration for Probabilistic Multi-Class Classification (MC-CLF) This section addresses probabilistic multi-class classification (MC-CLF), where the target variable takes values from a finite discrete set $\{c_1, \ldots, c_K\}$. Predicting a conditional CDF in this setting is equivalent to estimating a categorical probability distribution: $p_v = \mathbb{P}(Y = c_v \mid x)$, for $v \in \{1, \ldots, K\}$. Raw outputs from many ML models often do not represent well-calibrated probabilities, necessitating transformation via calibration methods.

Isotonic Regression Calibration Isotonic Regression (IR) [35, 33] is a non-parametric method which is a common baseline approach for MC-CLF. For binary classification ($K = 2$), IR uses a scoring function $s : \mathcal{X} \to [0, 1]$, which represents the uncalibrated probability estimate for the positive class. Given s, IR learns a non-decreasing step-function $f_{\text{IR}} : [0, 1] \to [0, 1]$ that maps a raw model score $s(x)$ to a calibrated probability $\hat{\mathbb{P}}(Y = 1 \mid x)$. This function is fitted to minimize error (e.g., squared loss) on a calibration dataset $\mathcal{D}_c = \{(x_j, y_j)\}_{j=1}^{N_c}$. Essentially, IR sorts instances by increasing scores, and finds the best-fitting monotonically increasing piecewise constant function (staircase). While useful, standard IR lacks finite-sample theoretical guarantees.

Inductive Venn-Abers Predictor (IVAP) As a theoretically robust alternative, *Venn-Abers Predictors* (VAP) leverage ideas from IR and the conformal prediction framework [29, 17]. VAPs offer distribution-free, finite-sample calibration guarantees under the mild assumption of exchangeability. The Inductive VAP (IVAP) modifies the standard IR approach to produce a probability *interval*, explicitly quantifying uncertainty. The IVAP procedure for binary classification (K = 2) is specified as follows:

1. **Data Splitting:** Divide the training dataset $\mathcal{D}_{\text{tr}}$ into two subsets: a proper training set $\mathcal{D}_t$ and a calibration set $\mathcal{D}_c = \{(x_j, y_j)\}_{j=1}^{N_c}$.
2. **Model Training:** Train a binary classification model on $\mathcal{D}_t$, yielding a scoring function $s : \mathcal{X} \to [0, 1]$, where $s(x)$ represents the uncalibrated probability estimate for the positive class.
3. **Dual Isotonic Regressions:** For a new instance x with score $s(x)$, construct the interval by fitting two separate IR models: f_0 and f_1. f_0, is fitted on $\{(s(x_j), y_j)\}_{j=1}^{N_c} \cup \{(s(x), 0)\}$, and f_1, is fitted on $\{(s(x_j), y_j)\}_{j=1}^{N_c} \cup \{(s(x), 1)\}$. These define the prediction interval

$$(p_0, p_1) := (f_0(s(x)), f_1(s(x)))\,,$$

which brackets the possible calibrated probabilities based on the two possible labels for the test instance. This dual fitting process prevents overly confident probability estimates. If the model is confident, p_0 and p_1 will be close together. A wider interval indicates greater uncertainty in the model output.

Cross-Validated Venn-Abers Predictor (CVAP) To enhance robustness over a single data split and obtain a single point estimate, the CVAP applies the IVAP procedure within a k-fold cross-validation scheme. This yields k probability intervals $[p_0^{(f)}, p_1^{(f)}]$ for a test instance. These intervals are then aggregated into a single probability estimate using a minimax-style geometric mean rule:

$$p = \frac{\text{GM}(p_1)}{\text{GM}(1-p_0) + \text{GM}(p_1)}, \tag{1}$$

where $\text{GM}(p_1) = \left(\prod_{f=1}^{k} p_1^{(f)}\right)^{1/k}$ and similarly for $\text{GM}(1-p_0)$. This aggregation balances information across all intervals and avoids overconfidence by integrating lower and upper bounds from each fold. The result is a single, calibrated probability estimate with strong empirical and theoretical support.

Extension to Multi-Class Classification Venn-Abers calibration is extended to multi-class problems ($K > 2$) using a one-versus-rest strategy. This involves training K distinct binary classifiers, where the i-th classifier distinguishes class i from all other classes. These binary classifiers are then individually calibrated with CVAP, yielding K probability estimates $p^1, \ldots, p^K$. As these are generated independently, a final normalization step ensures they form a valid probability distribution: $\hat{p}_v = p^v / \sum_{v'=1}^{K} p^{v'}$, for each class $v \in \{1, \ldots, K\}$.

4.3 A Two-Stage Method: Classify-then-Regress with Truncated Conformal Prediction

This section proposes a two-stage forecasting method for modeling fulfillment time deviations that combines *Venn-Abers-calibrated classification* with *conformal predictive system for regression*. The method is specifically designed to handle the mixed nature of the deviation variable: many orders arrive exactly on time, while the rest exhibit continuous variability in early or late arrivals. The proposed method—referred to as the *truncation-based classify-then-regress approach*—generates a full probability distribution over the delivery deviation for each order. It effectively integrates a categorical structure by first classifying orders into three distinct groups: early, on-time, or late. Within each of these categories, it then captures the continuous variability by predicting the specific magnitude of deviation. To ensure reliable probabilistic estimates, the approach incorporates conformal regression and Venn-Abers predictors. This combination enables both interpretable classification and precise, well-calibrated regression predictions.

Step 1: Classification Stage — Predicting Delivery Status Define a discrete random variable $S_i \in \{-1, 0, 1\}$ representing the delivery status:

$$S_i = \begin{cases} -1 & \text{if } Y_i < 0 \quad \text{(early)}, \\ 0 & \text{if } Y_i = 0 \quad \text{(on-time)}, \\ 1 & \text{if } Y_i > 0 \quad \text{(late)}. \end{cases}$$

A multi-class classifier is trained and calibrated using the Venn-Abers procedure, returning the class probabilities $P_c(S = s \mid x),$ for each $s \in \{-1, 0, 1\}$.

Step 2: Regression Stage — Predicting Deviation Magnitude with Truncation A conformal regression model (SCPS or MCPS, see Section 4.2) is trained on the full dataset. For each calibration example (x_j, y_j), define the calibration score for a test input x with model prediction $h(x)$ as $C_j = h(x) + (y_j - h(x_j))$. These scores simulate realistic outcomes by combining model predictions with observed residuals. Let the sorted scores be $C_{(1)} < C_{(2)} < \cdots < C_{(N_c)}$, and identify the index m and n such that $m = \max\{m : C_{(m)} < 0\}$ and $n = \min\{n : C_{(n)} > 0\}$. At prediction time, the following truncation strategy is applied to generate class-consistent predictive distributions:

- **On-time deliveries** ($S = 0$): Define $P_r(y \mid x, S = 0)$ as a degenerate distribution concentrated at 0. That is, the model assigns all probability mass to the outcome $y = 0$.
- **Early deliveries** ($S = -1$): Construct the predictive distribution $P_r(y \mid x, S = -1)$ using only scores $\{C_{(1)}, \ldots, C_{(m)}\}$, i.e., scores corresponding to negative deviations.
- **Late deliveries** ($S = 1$): Construct the distribution $P_r(y \mid x, S = 1)$ with scores $\{C_{(n)}, \ldots, C_{(N_c)}\}$, i.e., scores greater than zero.

This truncation ensures consistency between the predicted status and the continuous deviation forecast, without discarding training data as in class-specific regression models.

Step 3: Mixture Formation — Integrating Classification and Regression Before combining the components, recall that each class-conditional distribution $P_r(y \mid x, S = s)$ captures how far off a delivery might be, assuming the shipment falls into class s. The classifier returns a probability for each class. To get the full predictive distribution, average the class-conditional distributions using the class probabilities as weights:

$$\hat{F}(y \mid x) = \sum_{s \in \{-1,0,1\}} P_c(S = s \mid x) \cdot P_r(y \mid x, S = s).$$

Alternative Strategy and Justification A straightforward alternative would be to train separate regression models and apply CPS calibration within each class. However, this strategy underperforms in practice due to *selection bias*: classification errors cause mismatches between training and prediction regimes, especially near class boundaries. The truncation-based approach addresses the limitations of separate regression models by using the full dataset during training and calibration, rather than dividing data based on predicted classes. This avoids the selection bias introduced by classification errors, which often lead to mismatches between training and prediction conditions near class boundaries. Instead of applying class-specific constraints during training, the truncation-based method enforces these constraints only at inference time. This strategy enhances robustness and ensures that calibration guarantees are preserved across the full range of predictions.

4.4 Obtaining Point Predictions from Distributional Forecasts

While the proposed framework focuses on generating full predictive distributions for fulfillment time deviations, many logistics systems continue to operate with single-point estimates for decision-making tasks. Moreover, point predictions are necessary for comparing the performance of distributional models against legacy systems that do not support uncertainty quantification. This section outlines several methods for deriving point predictions from distributional forecasts, ranging from standard model-specific outputs to cost-sensitive decision rules tailored to real-world priorities.

Default Point Prediction Methods For CPS-based methods, a straightforward and principled choice is the median (50% quantile) of the predicted CDF. This corresponds to the point that minimizes expected absolute error. For MC-CLF models, the predicted class with the highest probability is selected: $\hat{y} = \arg\max_j \mathbb{P}(Y = c_j \mid x)$, representing the most probable delivery day.

Cost-Sensitive Decision Rule To go beyond fixed quantiles, this paper introduces a cost-sensitive decision rule for selecting a quantile that minimizes a weighted loss function reflecting practical trade-offs. The procedure is:

1. Fit CPS and/or CVAP models to obtain predictive distributions.
2. On a validation set, perform a grid search over $\eta \in [0, 100]$.
3. For each candidate η, extract point predictions $\hat{y}^\eta$ as the η%-quantile of the CDF.
4. Evaluate the following cost-sensitive objective:

$$\min_{\eta \in [0,100]} \text{RMSE}(\hat{y}^\eta) + \beta\ \text{Late_RMSE}(\hat{y}^\eta) + \gamma\ \text{Early_RMSE}(\hat{y}^\eta),$$

where RMSE is the overall root mean squared error, Late_RMSE corresponds to the RMSE over late deliveries, and Early_RMSE captures the RMSE for early deliveries. The penalty weights γ and β reflect user-defined preferences for penalizing early or late deliveries. The optimal quantile η^* is then applied to each test instance i to produce final point predictions:

$$\hat{y}_i = \inf \left\{ y : \hat{\mathbb{P}}(Y_i \leq y \mid x_i) \geq \eta^* \right\}.$$

This formulation enables practitioners to tailor prediction strategies according to operational priorities, such as minimizing customer dissatisfaction due to early or late arrivals.

5 Experiment Results

This section provides experimental results on the proposed methods. The evaluation uses large-scale industrial instances, and the proposed methods are compared with several baseline methods.

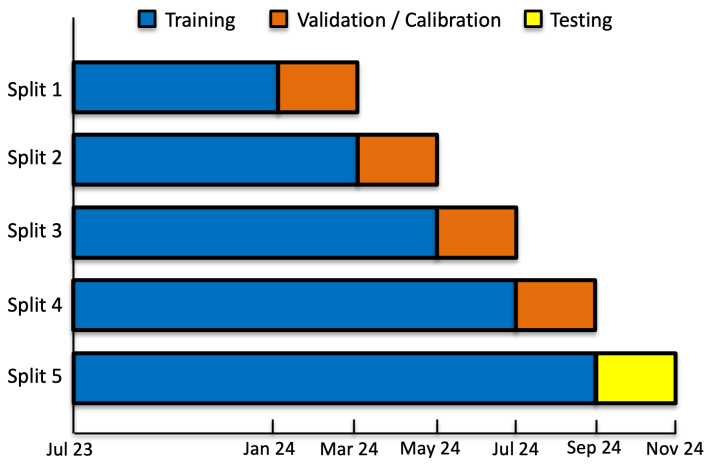


Fig. 4: Data Splitting for the Experimental Results.

5.1 The Experimental Setting

Data Splitting Organization The dataset was partitioned using a 4-fold time-series cross-validation approach to preserve temporal dependencies, as illustrated in Fig. 4. The validation sets were consistently used across all training runs to tune the hyperparameters of the ML models. For CVAP, calibration sets from all four folds were utilized, whereas for SCPS and MCPS, only the calibration set from the last fold was used. To ensure a fair evaluation, the last two months of data were reserved for testing. Within the test set, only orders that were shipped via the same carrier service that was estimated at the time of location selection were retained. This restriction ensures a direct and unbiased comparison between the proposed models and the existing system.

Tools and Models The CPS-based models were implemented using Crepes [4], while the CVAP was adapted from the Venn-ABERS Calibration library. For the MCPS, the number of bins was set to 10 to balance between model flexibility and computational efficiency. XGBoost [8] was used as the underlying regression model for all CPS methods, while CatBoost [22] was employed for MC-CLF. Both models are widely adopted in industry and demonstrated superior performance in initial experiments compared to other standard machine learning models, aligning with the findings in [33]. Optuna was used for hyperparameter tuning for all ML models [1].

Feature Importance Interpretability plays a crucial role in the deployment of ML models in industrial applications. Tree-based ML models inherently provide feature importance scores by aggregating the gains from feature splits. Fig. 5 illustrates the top 20 most influential features identified by the XGBoost regression model. Order-level features rank highest in importance, but route representation and FL operation features also significantly impact predictions. Key route-related features include average transit time between hubs and frequently used hub-to-hub distances, while FL operation features like wave time segment and processing time contribute to fulfillment time variability. These findings confirm the necessity of incorporating both route and FL operation representations in effective fulfillment time forecasting.

Evaluation Metrics Table 1 summarizes the evaluation metrics considered in the experiments. In particular, the continuous ranked probability score (CRPS) measures the accuracy of the probability density forecast. The pinball loss (PL) and the mean absolute quantile coverage error (MQCE) measure the sharpness and reliability of the quantile estimates. The accuracy, RMSE, and the mean absolute error (MAE) evaluate the point predictions. The accuracy measures the number of times that an order is delivered exactly on the predicted date. RMSE and MAE measure the differences in days between the predicted and the correct dates.

5.2 Distributional Forecasting Performance

Table 2 summarizes the distributional forecasting performance on the test set: it reports the PL, MQCE, CRPS, and the coverage and mean interval sizes for three symmetric prediction intervals (80%, 90%, 95%). The implementations

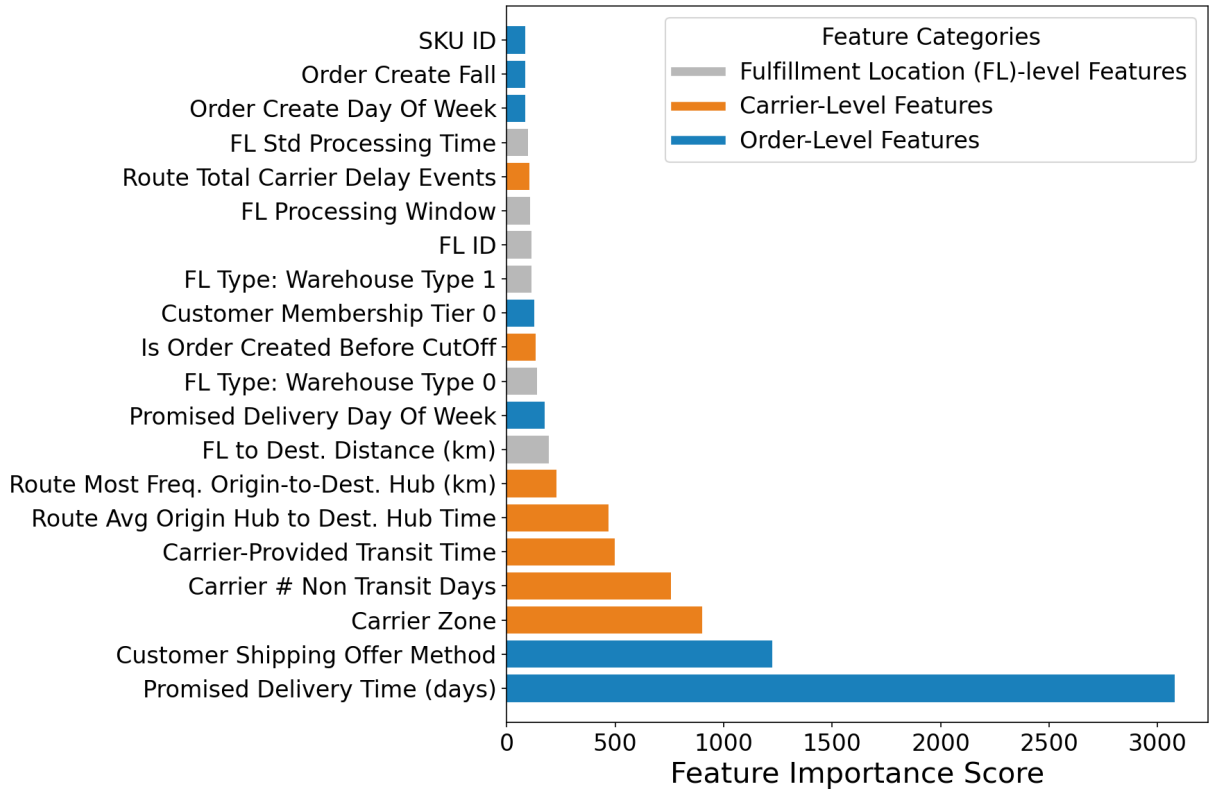


Fig. 5: The Top 20 Most Important Features used by the XGBoost Regressor.

Table 1: Summary and Description of Evaluation Metrics.

Metric	**Formula**	**Description**
CRPS	$\frac{1}{N_{test}} \sum_{i=1}^{N_{test}} \int_{-\infty}^{\infty} (\hat{F}(u) - \mathbf{1}\{u \geq y_i\})^2 du$	Continuous ranked probability score.
PL	$\frac{1}{9} \sum_{q \in [0.1, 0.2, \ldots, 0.9]} \frac{1}{N_{test}} \sum_{i=1}^{N_{test}} \Big(q \max\{y_i - \hat{y}_i^{100q}, 0\} + (1-q) \max\{\hat{y}_i^{100q} - y_i, 0\} \Big)$	Average pinball loss.
MQCE	$\frac{1}{9} \sum_{q \in [0.1, 0.2, \ldots, 0.9]} \lvert \{ \frac{1}{N_{test}} \sum_{i=1}^{N_{test}} \mathbf{1}_{y_i < \hat{y}_i^{100q}} \} - q \rvert$	Mean quantile coverage error.
Accuracy	$\frac{1}{N_{test}} \sum_{i=1}^{N_{test}} \mathbf{1}(y_i = \hat{y}_i)$	Number of correct predictions over the total number of predictions.
RMSE	$\sqrt{\frac{1}{N_{test}} \sum_{i=1}^{N_{test}} (\hat{y}_i - y_i)^2}$	Root mean squared error (in number of days).
MAE	$\frac{1}{N_{test}} \sum_{i=1}^{N_{test}} \lvert \hat{y}_i - y_i \rvert$	Mean absolute error (in number of days).

of Quantile Random Forest (QRF) and Isotonic Regression (IR)-calibrated MC-CLF closely follow the modeling approaches in [33]. Fig. 6 shows the prediction intervals generated by the regression-based and two-stage methods.

Overall, the calibrated MC-CLF models outperform their uncalibrated counterparts in CRPS by around 16%. The CVAP calibration shows competitive performance with IR calibration, with marginally better PL and MQCE. Notably CVAP yields higher coverage rates, though neither calibration approach perfectly matches the targeted coverage levels at all intervals. While calibrated classifiers excel in distribution accuracy, CPS-based regression methods demonstrate superior quantile estimate reliability through lower pinball loss (5% to 10% lower than QRF) and better interval coverages (all reaching nominal coverages). MCPS produces on average 0.1 to 0.3 days narrower intervals than SCPS with stronger distributional metrics due to its adaptive binning.

Table 2: Average Pinball Loss, MQCE, CRPS, Coverage, and Interval Sizes on the Test Set.

Methods	**PL ↓**	**MQCE ↓**	**CRPS ↓**	**Coverage**			**Mean Interval Size ↓**		
				80%	90%	95%	80%	90%	95%
Base MC-CLF	0.335	0.232	0.498	0.427	0.600	0.768	2.043	2.749	3.517
IR MC-CLF	0.298	0.240	**0.413**	0.619	0.776	0.901	2.232	2.944	3.659
CVAP MC-CLF	0.297	0.232	**0.419**	0.658	0.835	0.935	2.312	3.049	3.780
QRF	0.273	0.313	0.502	0.876	0.923	0.965	**1.844**	**2.587**	**3.445**
SCPS	0.261	**0.035**	0.482	0.817	0.907	0.953	2.133	3.374	4.582
MCPS	**0.244**	**0.041**	0.451	0.829	0.915	0.953	2.097	3.002	4.200
2stg-SCPS	**0.238**	0.217	**0.440**	0.795	0.879	0.935	**1.439**	**2.264**	**3.172**
2stg-MCPS	**0.242**	**0.213**	0.449	0.828	0.886	0.925	**1.664**	**2.391**	**3.135**

*Note: ↓ indicates lower values are better. Best three values are **bolded**.*
Coverage values that achieve nominal rates (80%, 90%, 95%) are underlined.

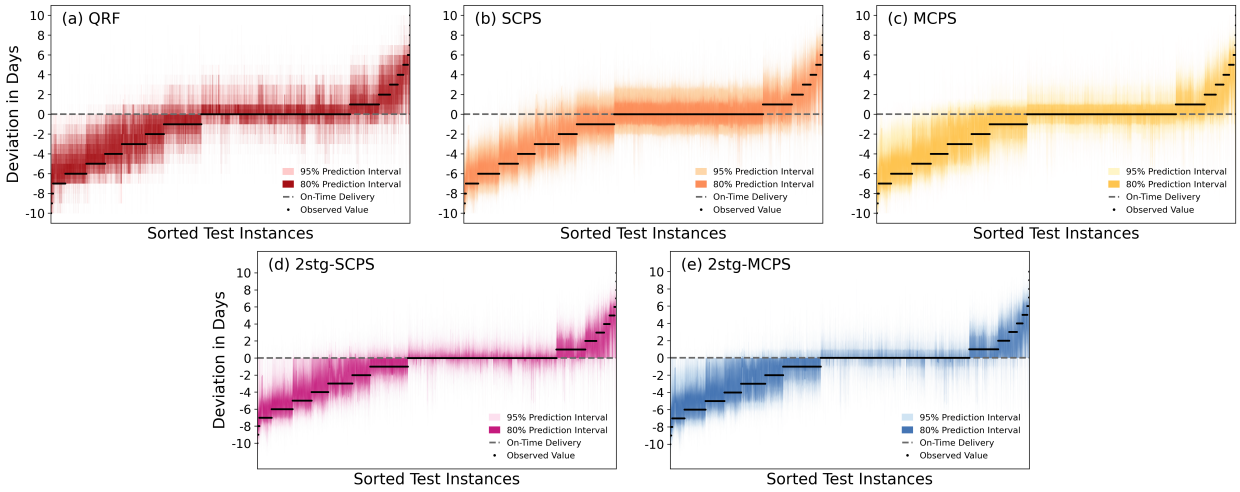


Fig. 6: Prediction Intervals on Sorted Test Set Instances.

Two-stage methods (2stg-SCPS/MCPS) combine these strengths, achieving remarkable sharpness. As shown in Fig. 6(d–e), their 80%, 90% and 95% bands are visibly the tightest of all – on average 0.2–0.4 days narrower than those from QRF in panel (a) – yet they do so without "wasting" coverage on the wrong side of zero. For the earliest deliveries (left of the zero-error line), virtually all of the interval mass lies strictly below zero, while for the latest shipments (right of zero) the bands shift are entirely above. The on-time cases densely cluster around zero with very narrow intervals, delivering pinpoint estimates of punctual performance. This targeted contraction translates directly into better scores: up to an 8.7% reduction in CRPS versus single-stage CPS and an 11–13% improvement in pinball loss relative to QRF, all while producing the narrowest uncertainty intervals overall. The trade-off is a modest dip in marginal coverage and a slightly looser quantile calibration compared to full CPS; in practice, it dramatically cuts unnecessary buffer days and yields far more actionable forecasts.

5.3 Point Prediction Performance

This section evaluates the point prediction performance of each model. The objective is to produce a point prediction that can be directly compared to the actual fulfillment date.

Methods for Generating a Single Estimated Delivery Date The methods used to derive a single estimated delivery date can be summarized as follows:

- *Current System*: The point prediction of the existing system which is a rule-based system using static transit time tables and cutoff times.
- *MC-CLF (Base/IR/CVAP)*: The point prediction is given by the MC-CLF model predicting the most likely day-offset.
- *CVAP MC-CLF-Rule*: A MC-CLF model calibrated by CVAP, which uses a cost-sensitive decision rule for quantile selection before rounding to the nearest day.
- *Base Reg*: The point prediction is obtained by a standard regression predicting the mean deviation, rounded to the nearest day.
- *SCPS/MCPS-Median*: The point prediction is the median of the CPS rounded to the nearest day.
- *SCPS/MCPS-Rule*: Applies cost-sensitive decision rule on the CPS for quantile selection before rounding to the nearest day.
- *2-stg-SCPS/MCPS-Rule*: The point prediction is obtained by applying a cost-sensitive decision rule on the two-stage model for quantile selection before rounding to the nearest day.

For all models using a cost-sensitive decision rule, the default parameter values are $\beta = 0.5$ and $\gamma = 0$. These settings prioritize the accurate prediction of late deliveries over early ones, reflecting a common operational objective, i.e., the minimization of late arrivals, which typically incur higher penalties or customer dissatisfaction.

Key Observations Table 3 reports the point prediction metrics, highlighting the overall accuracy, RMSE, MAE, the percentage of late deliveries identified, and error metrics specifically on late deliveries.

The current system's sub-50% accuracy reveals significant room for data-driven improvement, particularly in identifying late deliveries. This limitation is unsurprising, given that the test set reflects historical decisions where the chosen fulfillment location was already deemed acceptable. As a result, the system is biased toward on-time predictions and performs poorly under dynamic or uncertain conditions.

In contrast, ML-based approaches enhance the overall accuracy by up to 14%, identify substantially more late deliveries (up to 75%), and reduce associated error metrics by up to 0.6 days overall and up to 1.5 days on actual late deliveries. Both IR and CVAP improve overall accuracy by around 5% compared to their uncalibrated version. The uncalibrated MC-CLF performs poorly on most metrics relative to other ML methods. By contrast, the calibrated MC-CLF models strike a better balance between accuracy and late-delivery detection (up to 13% higher than Base MC-CLF). CPS methods offer stronger calibration for point predictions than a simple regression baseline. Indeed, comparing SCPS-/MCPS-Median with Base Reg reveals gains in overall accuracy (2% to 3% higher) and error metrics (0.01 to 0.03 days lower). However, their performance on strictly late deliveries can be less clear-cut. Since calibration primarily refines the typical (mean or median) estimation, it may not always help with the tail behavior tied to late arrivals.

Introducing a cost-sensitive decision rule—particularly SCPS- and MCPS-Rule—shifts the emphasis toward more aggressively predicting late shipments. This leads to higher late-detection rates (1.5% to 4.2% higher than Base Reg) and lower late-delivery errors (up to 0.05 days lower than Base Reg), albeit at the cost of overall accuracy (4% to 8% lower than Base Reg). As shown in Fig. 7, the degree of this trade-off can be fine-tuned by adjusting the weight β in the decision rule's objective function, allowing the system to balance penalties for late misclassification against the desire for a high overall prediction accuracy. When applied to two-stage methods however, these decision rules demonstrate reduced effectiveness. This diminished performance likely stems from the two-stage methods' inherent prioritization of tighter prediction intervals over quantile estimation reliability.

6 Discussion and Managerial Implications

The value of distributional forecasting extends beyond statistical accuracy to the quality of the downstream decisions it enables. By quantifying the full spectrum of potential delivery outcomes, the proposed framework allows managers to proactively manage risk. The optimal forecasting model, however, is contingent on the specific managerial objective.

Informing Stochastic and Robust Optimization For risk-neutral stochastic optimization, where the goal is to optimize expected outcomes, CVAP MC-CLF is highly valuable. It produces calibrated distributional forecasts with validity guarantees, which can inform decisions such as selecting a carrier or fulfillment location [33]. For risk-averse or robust optimization, SCPS/MCPS methods are more suitable, generating prediction intervals with strong coverage

Table 3: Point Prediction Metrics on the Test Set.

Methods	Overall Accuracy(%) ↑	Overall RMSE ↓	Overall MAE ↓	Late Detection(%) ↑	Late RMSE ↓	Late MAE ↓
Current System	49.53	1.66	1.01	3.82	2.99	2.49
Base MC-CLF	58.50	1.30	0.71	48.67	1.84	1.31
IR MC-CLF	**64.41**	1.14	0.58	63.60	1.71	1.09
CVAP MC-CLF	**64.14**	1.16	0.59	62.53	1.72	1.11
CVAP MC-CLF-Rule	52.47	1.30	0.77	73.60	1.62	1.09
Base Reg	58.79	**1.01**	**0.56**	**75.17**	**1.52**	**0.97**
SCPS-Median	60.23	**1.00**	**0.54**	73.37	1.54	0.98
MCPS-Median	**61.47**	**1.00**	**0.53**	63.21	1.60	1.07
SCPS-Rule ($\beta = 0.5$)	54.32	1.05	0.62	**79.37**	**1.48**	**0.95**
MCPS-Rule ($\beta = 0.5$)	50.63	1.12	0.68	**76.66**	**1.47**	**0.93**
2stg-SCPS-Rule	58.83	1.08	0.6	66.71	1.56	1.04
2stg-MCPS-Rule	58.18	1.08	0.6	59.99	1.59	1.08

Note: Arrows indicate whether higher (↑) or lower (↓) values are better.
Bold numbers indicate top three values for each metric.

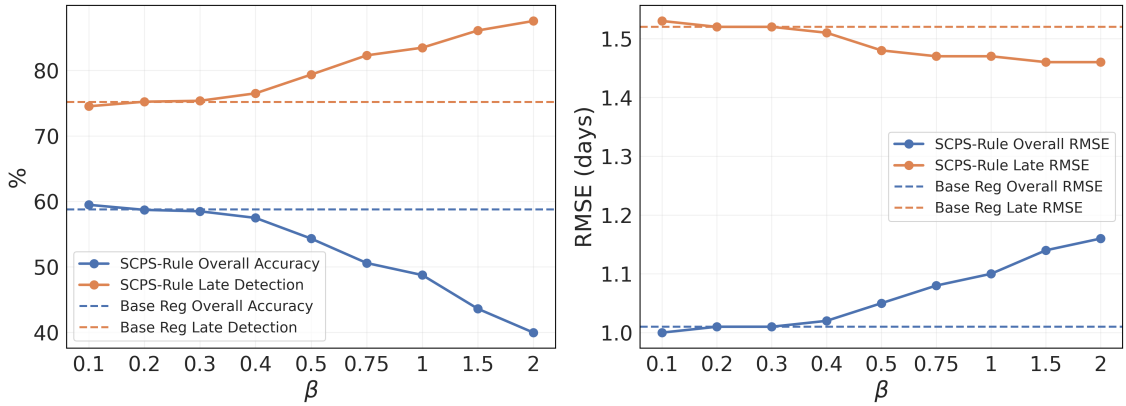


Fig. 7: Accuracy / RMSE vs. Late Delivery Weight (β) in the Decision Rule Objective.

guarantees. These intervals can construct the uncertainty sets over which a decision is robustified. This is crucial for applications like establishing reliable delivery windows that serve as core constraints for vehicle routing, or guiding strategic network decisions, such as identifying high-performance warehouses to ensure sufficient inventory allocation.

Integrating with Operational Workflows For direct integration into rule-based operational workflows, the choice of model depends on system capabilities. Two-stage models produce more actionable intervals; by first predicting a delivery status (e.g., late), they then generate a deviation interval conditional on that outcome, which facilitates transparent triage rules. For legacy systems requiring point forecasts, SCPS/MCPS-Median offers strong risk-neutral accuracy. For cost-sensitive decisions where late deliveries must be minimized, SCPS/MCPS-Rule provides targeted control, with a tunable β weight to flexibly manage the trade-off. This is directly applicable to proactive customer communication, where a conservative forecast can trigger early notifications, or staffing decisions, where anticipated delays can justify scheduling contingent labor.

7 Conclusions

This study presents a novel framework for distributional forecasting of delivery times, leveraging conformal prediction techniques to provide robust and interpretable uncertainty estimates. By incorporating granular spatio-temporal features, the proposed approach enables more accurate and adaptable probabilistic predictions. The introduction of a cost-sensitive decision rule further ensures that point predictions align with business objectives, balancing accuracy and the detection of late deliveries.

Importantly, the industrial partner has expressed strong interest in deploying something very similar to the proposed system. Components of this system are under active development to be tested against the current system. Initially, the proposed system will most likely be used to augment the existing location-selection process by providing delivery-time guidance alongside the current rule-based estimation infrastructure.

For future work, a key direction is integrating this framework with omnichannel fulfillment optimization (e.g., [33]) to evaluate their impact on downstream tasks, such as location and carrier selection. Fulfillment time forecasting with coverage or validity guarantees could provide provable bounds on the objective of the downstream optimization problem, enhancing decision-making reliability. In addition, there are several promising ways to extend the forecasting models. One approach is to integrate deep learning techniques with conformal predictive distributions. Another is to enrich the input space by incorporating additional features, such as autoregressive time-series components [27] or weather-related information [36].

Acknowledgments. This research was partly supported by the NSF AI Institute for Advances in Optimization (Award 2112533)

References

[1] Akiba, T., Sano, S., Yanase, T., Ohta, T., Koyama, M.: Optuna: A next-generation hyperparameter optimization framework. In: Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining. pp. 2623–2631 (2019)
[2] de Araujo, A.C., Etemad, A.: End-to-end prediction of parcel delivery time with deep learning for smart-city applications. IEEE Internet of Things Journal **8**(23), 17043–17056 (2021)
[3] Arora, S., Taylor, J.W., Mak, H.Y.: Probabilistic forecasting of patient waiting times in an emergency department. Manufacturing & Service Operations Management **25**(4), 1489–1508 (2023)
[4] Boström, H.: crepes: a python package for generating conformal regressors and predictive systems. In: Conformal and Probabilistic Prediction with Applications. pp. 24–41. PMLR (2022)
[5] Boström, H., Johansson, U., Löfström, T.: Mondrian conformal predictive distributions. In: Conformal and Probabilistic Prediction and Applications. pp. 24–38. PMLR (2021)
[6] Bruys, T., Zandehshahvar, R., Hijazi, A., Van Hentenryck, P.: Confidence-aware deep learning for load plan adjustments in the parcel service industry. arXiv preprint arXiv:2411.17502 (2024)
[7] Cao, J., Leng, Y., Wang, H.: Revenue-centered delivery time presentation on platforms: A spatial neural ode approach (2024)
[8] Chen, T., Guestrin, C.: Xgboost: A scalable tree boosting system. In: Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining. pp. 785–794 (2016)
[9] Cheng, S., Hijazi, A., Konak, J., Erera, A., Van Hentenryck, P.: Spot: Spatio-temporal pattern mining and optimization for load consolidation in freight transportation networks. arXiv preprint arXiv:2504.09680 (2025)
[10] Cui, R., Lu, Z., Sun, T., Golden, J.M.: Sooner or later? promising delivery speed in online retail. Manufacturing & Service Operations Management **26**(1), 233–251 (2024)
[11] Faulkner, S., Zandshahvar, R., Akhlaghi, V.E., Hentenryck, P.V.: A data-driven deep learning framework for package delivery time estimation and delay duration prediction (2025), Working Paper
[12] Guo, X., Grushka-Cockayne, Y., De Reyck, B.: Forecasting airport transfer passenger flow using real-time data and machine learning. Manufacturing & Service Operations Management **24**(6), 3193–3214 (2022)
[13] Jonkers, J., Avendano, D.N., Van Wallendael, G., Van Hoecke, S.: A novel day-ahead regional and probabilistic wind power forecasting framework using deep cnns and conformalized regression forests. Applied Energy **361**, 122900 (2024)
[14] Li, M., Klamkin, M., Tanneau, M., Zandehshahvar, R., Van Hentenryck, P.: Conformal prediction with upper and lower bound models. arXiv preprint arXiv:2503.04071 (2025)
[15] Li, Y., Wu, X., Wang, J., Liu, Y., Wang, X., Deng, Y., Miao, C.: Unsupervised categorical representation learning for package arrival time prediction. In: Proceedings of the 30th ACM International Conference on Information & Knowledge Management. pp. 3935–3944 (2021)
[16] Liu, H., Jiang, W., Liu, S., Chen, X.: Uncertainty-aware probabilistic travel time prediction for on-demand ride-hailing at didi. In: Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. pp. 4516–4526 (2023)
[17] Manokhin, V.: Multi-class probabilistic classification using inductive and cross venn–abers predictors. In: Conformal and Probabilistic Prediction and Applications. pp. 228–240. PMLR (2017)

[18] Manokhin, V.: Awesome conformal prediction (Apr 2022), https://doi.org/10.5281/zenodo.6467205
[19] Meinshausen, N., Ridgeway, G.: Quantile regression forests. Journal of machine learning research **7**(6) (2006)
[20] Merchán, D., Arora, J., Pachon, J., Konduri, K., Winkenbach, M., Parks, S., Noszek, J.: 2021 amazon last mile routing research challenge: Data set. Transportation Science **58**(1), 8–11 (2024)
[21] Moradi, A., Tanneau, M., Zandehshahvar, R., Van Hentenryck, P.: Enhanced renewable energy forecasting and operations through probabilistic forecast aggregation. arXiv preprint arXiv:2502.07010 (2025)
[22] Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A.V., Gulin, A.: Catboost: unbiased boosting with categorical features. Advances in neural information processing systems **31** (2018)
[23] Raj, G., Roy, D., De Koster, R., Bansal, V.: Stochastic modeling of integrated order fulfillment processes with delivery time promise: Order picking, batching, and last-mile delivery. European Journal of Operational Research **316**(3), 1114–1128 (2024)
[24] Ruan, S., Long, C., Ma, Z., Bao, J., He, T., Li, R., Chen, Y., Wu, S., Zheng, Y.: Service time prediction for delivery tasks via spatial meta-learning. In: Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. pp. 3829–3837 (2022)
[25] Salari, N., Liu, S., Shen, Z.J.M.: Real-time delivery time forecasting and promising in online retailing: When will your package arrive? Manufacturing & Service Operations Management **24**(3), 1421–1436 (2022)
[26] Shang, Y., Dunson, D., Song, J.S.: Exploiting big data in logistics risk assessment via bayesian nonparametrics. Operations Research **65**(6), 1574–1588 (2017)
[27] Tang, A., Halmkrona Lahtinen, J.: Applying software engineering principles to develop parcel delay forecasting models using tracking data: A study of the models ARIMA, BSTS, and GAM. Master's thesis, University of Gothenburg (2022)
[28] Vovk, V., Gammerman, A., Shafer, G.: Algorithmic learning in a random world, vol. 29. Springer (2005)
[29] Vovk, V., Petej, I., Fedorova, V.: Large-scale probabilistic predictors with and without guarantees of validity. Advances in Neural Information Processing Systems **28** (2015)
[30] Vovk, V., Petej, I., Nouretdinov, I., Manokhin, V., Gammerman, A.: Computationally efficient versions of conformal predictive distributions. Neurocomputing **397**, 292–308 (2020)
[31] Wen, H., Lin, Y., Wu, L., Mao, X., Cai, T., Hou, Y., Guo, S., Liang, Y., Jin, G., Zhao, Y., et al.: A survey on service route and time prediction in instant delivery: Taxonomy, progress, and prospects. IEEE Transactions on Knowledge and Data Engineering (2024)
[32] Wu, F., Wu, L.: Deepeta: a spatial-temporal sequential neural network model for estimating time of arrival in package delivery system. In: Proceedings of the AAAI conference on artificial intelligence. vol. 33, pp. 774–781 (2019)
[33] Ye, T., Cheng, S., Hijazi, A., Van Hentenryck, P.: Contextual stochastic optimization for omnichannel multi-courier order fulfillment under delivery time uncertainty. arXiv preprint arXiv:2409.06918 (2024)
[34] Yi, J., Yan, H., Wang, H., Yuan, J., Li, Y.: Learning to estimate package delivery time in mixed imbalanced delivery and pickup logistics services. In: Proceedings of the 32nd ACM International Conference on Advances in Geographic Information Systems. pp. 432–443 (2024)
[35] Zadrozny, B., Elkan, C.: Transforming classifier scores into accurate multiclass probability estimates. In: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 694–699 (2002)
[36] Zhang, H., Zandehshahvar, R., Tanneau, M., Van Hentenryck, P.: Weather-informed probabilistic forecasting and scenario generation in power systems. Applied Energy **384**, 125369 (2025). https://doi.org/https://doi.org/10.1016/j.apenergy.2025.125369, https://www.sciencedirect.com/science/article/pii/S0306261925000996
[37] Zhang, L., Wu, X., Liu, Y., Zhou, X., Cao, Y., Xu, Y., Cui, L., Miao, C.: Estimating package arrival time via heterogeneous hypergraph neural network. Expert Systems with Applications **238**, 121740 (2024)
[38] Zhang, L., Zhou, X., Zeng, Z., Cao, Y., Xu, Y., Wang, M., Wu, X., Liu, Y., Cui, L., Shen, Z.: Delivery time prediction using large-scale graph structure learning based on quantile regression. In: 2023 IEEE 39th International Conference on Data Engineering (ICDE). pp. 3403–3416. IEEE (2023)

Appendix

Table 4: Features Used in the Machine Learning Model across Order, Fulfillment Location, and Carrier Levels.

Feature Name	Description	Type
Order-Level Features		
Order Create Month/Day/Hour	Date-time components when order was created	Numerical
Order Create Season/Holiday	Season and holiday indicator of order date	Categorical/Binary
Promised Delivery Month/Day	Date-time components of promised delivery	Categorical
Promised Delivery Holiday	Whether promised date is a holiday	Binary
Promised Delivery Time	Days from order to promised delivery	Numerical
Shipping Offer Method	Fulfillment experience at order time	Categorical
Order Total Units, Weight, Dim. Weight	Size and quantity metrics of the order	Numerical
SKU ID/Units/Height/Length/Width/Weight	SKU attributes and quantity in order	Mixed
SKU Dept/Class/Subclass/Perf. Code	SKU category and performance level	Categorical
Customer Membership Tier	Customer membership type	Categorical
Dest. ZIP/State	Customer location	Categorical
Fulfillment Location (FL)-Level Features		
FL ID/Type/Lat/Lon	Fulfillment location identifier and position	Categorical/Numerical
FL to Dest. Distance	Distance to customer	Numerical
SKU On-Hand Inventory	SKU availability at location	Numerical
FL Capacity Status	Whether FL is over capacity	Binary
FL Waiting Units / Same SKU	Queue length for all and same SKU	Numerical
FL Shipped Units Last 2Hrs / Same SKU	Recent shipping activity	Numerical
FL Processing Window	Time window of order creation	Categorical
FL Avg/Std Processing Time	Processing time stats	Numerical
Carrier-Level Features		
Transit Time / Zone / Cutoff Time	Carrier time and region details	Numerical
# Non Transit Days / Before Cutoff	Service schedule and timing compliance	Mixed
Route Use Freq. / Median # Stops/Updates	Route-level shipment frequency	Numerical
Total Carrier/Weather/Other Exceptions	Exception event counts	Numerical
Avg/Std: Label to Hub, Hub Induction, Hub Transit	Timings between key shipping stages	Numerical
Avg/Std: Dest. Scan to Delivery, Final Delivery	Delivery segment durations	Numerical
Most Freq. Origin/Dest. Hub	Common route endpoints	Categorical
Most Freq. Hub Lat/Lon, Distance	Geolocation and distance between hubs	Numerical