Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

When to Contact the Buyer? An Explainable Survival Framework for Sales Intervention Prioritization

Joona Mäntyvaara

joeman@utu.fi

University of Turku

Tanja Vähämäki

University of Turku

Paavo Nevalainen

University of Turku

Jukka Heikkonen

University of Turku

Research Article

Keywords: Survival analysis, DeepSurv, SHAP, B2B sales, Buying window, Sales timing

Posted Date: November 26th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-10743294/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

When to Contact the Buyer? An Explainable Survival Framework for Sales Intervention Prioritization

Joona Mäntyvaara[1*], Tanja Vähämäki[1], Paavo Nevalainen[1], Jukka Heikkonen[1]

[1]University of Turku, Department of Computing, Turku, Finland.

*Corresponding author(s). E-mail(s): joeman@utu.fi;
Contributing authors: tmvaha@utu.fi; ptneva@utu.fi; jukhei@utu.fi;

Abstract

Identifying the optimal moment for sales intervention in B2B contexts remains challenging: contacting customers too early risks rejection, while contacting too late may result in lost sales. This paper proposes a survival-based framework for detecting "pre-buying windows"—periods when multiple hazard-driving signals converge, indicating that a customer is transitioning toward purchase receptivity—in industrial spare-part sales. The framework combines Random Survival Forest for feature selection, DeepSurv for non-linear survival prediction, and SHAP-based convergence detection. The convergence score quantifies how many feature-level SHAP values simultaneously exceed their respective thresholds, flagging customers whose buying windows may be opening. The framework is validated on industrial transaction data comprising 523 repeat customers and 12,970 transactions. The DeepSurv model achieves a concordance index of 0.735 on held-out test data, and convergence analysis identifies 62 customers (11.9%) with opening buying windows. Ablation analysis reveals that customer tenure is the strongest predictor, while purchase cycle features provide interpretable dimensions for sales engagement. The framework outputs a prioritized list of flagged customers with their elevated features, enabling direct application to sales operations.

Keywords: Survival analysis, DeepSurv, SHAP, B2B sales, Buying window, Sales timing

1 Introduction

Timing is widely recognized as an important factor in B2B sales effectiveness [1–3]. Sales teams must allocate limited capacity across many potential contacts, and mistimed interventions create measurable opportunity cost. Contacting a customer too early may result in rejection due to a lack of immediate need, while contacting too late risks losing the sale to competitors or allowing the purchase window to close entirely [4]. In industrial spare-part sales, where customers maintain equipment with varying maintenance cycles and operational demands, identifying the optimal moment for sales intervention presents a significant challenge.

Traditional approaches to sales timing often rely on simple heuristics such as fixed recontact intervals or recency-based triggers [5]. While these methods are easy to implement, they fail to account for customer heterogeneity and the complex interplay of factors that influence purchase decisions. More sophisticated predictive models can estimate purchase probability, but they typically provide a single score that offers limited insight into *why* a customer is likely to purchase or *when* the optimal intervention moment occurs.

It is hypothesized that the optimal timing for a sales intervention corresponds to entering a "buying window" — a period when multiple purchase-driving signals converge. Rather than relying on any single indicator, it is proposed that the convergence of elevated signals across multiple dimensions (behavioral patterns, financial indicators, and purchase cycle timing) indicates that a customer is transitioning from low to high purchase receptivity. This convergence reflects a transition toward a buying window, indicating a potential opportunity for timely sales engagement.

To operationalize this hypothesis, a three-stage framework is developed combining survival analysis with machine learning interpretability:

1. **Feature selection** using Random Survival Forest [6] to identify predictive behavioral features, augmented with timing features that capture purchase cycle dynamics.
2. **Survival prediction** using DeepSurv [7], a neural network extension of the Cox proportional hazards model, to learn non-linear relationships between customer features and reorder hazard.
3. **Convergence detection** using SHAP [8] (SHapley Additive exPlanations) to decompose predictions into per-feature contributions and identify customers where multiple features are simultaneously elevated above their respective thresholds.

The main contribution of this work is a convergence-based framework for detecting buying windows in B2B sales contexts. Unlike prior survival prediction approaches that focus primarily on estimating purchase probability, the proposed approach leverages SHAP decomposition to identify when multiple hazard-driving signals simultaneously exceed their respective thresholds. A convergence score is introduced to quantify how many features contribute strongly to elevated hazard at a given moment. Customers with high convergence scores exhibit the "signal convergence" characteristic of an opening buying window, while the SHAP decomposition provides interpretable explanations identifying the specific factors driving each prediction.

The framework is validated on transaction data from a global industrial equipment manufacturer, demonstrating that the combination of customer profile and purchase cycle features achieves meaningful predictive performance (C-index = 0.735 on held-out test data) and that SHAP-based convergence detection identifies a subset of customers (11.9%) whose buying windows may be opening.

2 Related Work

Survival analysis has a rich history in marketing research, particularly for modeling customer lifetime value and purchase timing. The seminal Pareto/NBD model [9]

and its extensions [10] established probabilistic frameworks for predicting customer purchase behavior. Gupta et al. [11] provided a comprehensive review of customer lifetime value modeling approaches, noting the importance of understanding purchase timing for marketing interventions. In B2B contexts specifically, Tamaddoni Jahromi et al. [12] demonstrated that survival-based approaches can effectively model customer churn and retention, while recent work by Sanchez Ramirez et al. [13] showed that incorporating usage data improves B2B churn prediction accuracy.

The Cox proportional hazards model [14] has become foundational for survival analysis, modeling the instantaneous risk (hazard) of an event as a function of covariates. In marketing contexts, the "event" is typically a purchase, and covariates describe customer behavior and characteristics. However, traditional Cox regression assumes linear effects, limiting its ability to capture complex non-linear relationships in customer behavior data [7].

Recent advances have extended survival analysis with deep learning. DeepSurv [7] replaces the linear predictor in Cox regression with a neural network, enabling automatic learning of non-linear covariate effects. Lee et al. [15] proposed DeepHit for competing risks, while Kvamme et al. [16] compared neural network approaches for time-to-event prediction. Wang et al. [17] provided a comprehensive survey of machine learning approaches for survival analysis.

These methods achieve strong predictive performance, but complex machine learning models are often criticized for their limited interpretability, making it difficult to understand the reasoning behind individual predictions [18, 19]. In applied decision-making contexts such as B2B sales, where data-driven approaches are increasingly adopted [20], this lack of transparency can limit practical usefulness, as sales representatives need to understand *why* a customer is predicted to purchase soon in order to tailor their outreach effectively.

The interpretability challenge has motivated significant research in explainable AI. LIME [21] provides local explanations by fitting interpretable models around individual predictions. SHAP [8] offers theoretically grounded feature attributions based on Shapley values from cooperative game theory, with properties including local accuracy, missingness, and consistency.

Molnar et al. [22] surveyed the state of interpretable machine learning, noting the importance of model-agnostic methods that can explain any predictor. Chen et al. [23] studied how explanations affect human decision-making. In survival analysis contexts, SHAP values can decompose hazard predictions into per-feature contributions, revealing which factors drive risk for individual customers.

While previous work has addressed survival prediction and model interpretability separately, limited research has combined these for actionable business insights. Specifically, no prior work has leveraged SHAP decomposition of survival models to detect the *convergence* of multiple hazard-driving signals as an indicator of optimal intervention timing. The proposed framework addresses this gap by using SHAP values not just for post-hoc explanation, but as the primary mechanism for detecting when a customer's buying window is opening.

3 Methodology

This paper proposes a three-stage framework for buying window detection in B2B spare-part sales. The framework combines survival analysis with machine learning interpretability to identify when multiple purchase-driving signals converge, indicating optimal timing for sales intervention.

3.1 Framework Overview

The pipeline consists of the following stages:

1. **Stage 1: Feature Selection** — Train a Random Survival Forest (RSF) on customer-level features and compute permutation importance to identify the most predictive features for reorder timing.
2. **Stage 2: Survival Prediction** — Train a DeepSurv neural network on the selected features to model non-linear covariate effects on hazard.
3. **Stage 3: Buying Window Detection** — Apply SHAP decomposition to DeepSurv predictions to identify when multiple feature-level hazard contributions converge, signaling an opening buying window.

3.2 Stage 1: Feature Selection with Random Survival Forest

Random Survival Forest [6] extends the random forest algorithm to right-censored survival data, accommodating customers who have not yet reordered by the observation cutoff. Standard random forests are not designed for censored observations. RSF is used for feature selection rather than LASSO-Cox because RSF captures non-linear feature importance and interactions that linear methods would miss.

For a dataset with n customers, each characterized by feature vector $\mathbf{x}_i \in \mathbb{R}^p$, survival time T_i, and censoring indicator δ_i, RSF constructs an ensemble of survival trees. Each tree is grown on a bootstrap sample, with random feature subsets considered at each split.

Permutation importance is computed by measuring the decrease in concordance index (C-index) when each feature is randomly permuted:

$$\text{Importance}_j = \frac{1}{K} \sum_{k=1}^{K} \left[C(\mathbf{X}, \mathbf{y}) - C(\mathbf{X}^{(\pi_j)}, \mathbf{y}) \right] \tag{1}$$

where C denotes the concordance index (formally defined in Section 3.6), $\mathbf{X}^{(\pi_j)}$ denotes the feature matrix with the j-th column permuted, and $K = 10$ permutation repeats are used to estimate variance.

Features are selected using an elbow detection method on the ranked importance scores, ensuring a minimum of 5 features are retained.

3.3 Stage 2: Survival Prediction with DeepSurv

DeepSurv is employed rather than traditional Cox regression because customer purchasing behavior often exhibits non-linear covariate effects — for example, diminishing returns of order value on purchase propensity, or threshold effects where feature influence changes above certain values. DeepSurv maintains the interpretable proportional hazards structure while allowing flexible non-linear transformations of the input features.

While RSF performs well for feature selection, DeepSurv offers advantages for the prediction stage. The neural network produces a continuous risk score suitable for SHAP decomposition, and the proportional hazards structure ensures that SHAP values decompose the log-hazard additively across features.

DeepSurv [7] extends the Cox proportional hazards model by replacing the linear risk function with a neural network:

$$h(t|\mathbf{x}) = h_0(t) \exp(\phi(\mathbf{x}; \boldsymbol{\theta})) \tag{2}$$

where $\phi(\mathbf{x}; \boldsymbol{\theta})$ is a feedforward neural network with parameters $\boldsymbol{\theta}$.

The network is trained by minimizing the negative Cox partial log-likelihood:

$$\mathcal{L}(\boldsymbol{\theta}) = -\sum_{i:\delta_i=1} \left[\phi(\mathbf{x}_i) - \log \sum_{j \in \mathcal{R}(T_i)} \exp(\phi(\mathbf{x}_j)) \right] \tag{3}$$

where $\mathcal{R}(T_i)$ is the risk set at time T_i.

The architecture consists of two hidden layers with 64 and 32 units respectively, ReLU activations, batch normalization [24], and dropout [25] (rate 0.3) for regularization. Training uses the Adam optimizer [26] with learning rate 10^{-3}, weight decay 10^{-4}, and early stopping based on validation C-index.

3.4 Stage 3: Buying Window Detection via SHAP Convergence

SHAP decomposition is applied rather than relying on global feature importance because buying window detection requires *customer-level* explanations — understanding which features are elevated for this specific customer at this moment, not which features matter on average. SHAP provides additive, locally faithful explanations that enable per-customer threshold detection.

The core innovation of this approach is detecting when multiple hazard-driving signals converge simultaneously. For each customer, SHAP values [8] are computed to decompose the DeepSurv risk score into per-feature contributions:

$$\phi(\mathbf{x}_i) = \phi_0 + \sum_{j=1}^{p} \phi_j(\mathbf{x}_i) \tag{4}$$

where $\phi_j(\mathbf{x}_i)$ represents feature j's contribution to customer i's log-hazard.

The *convergence score* counts the number of features simultaneously contributing positively above a threshold:

$$\text{CS}_i = \sum_{j=1}^{p} \mathbf{1}\left[\phi_j(\mathbf{x}_i) > \tau_j\right] \tag{5}$$

where τ_j is the 75th percentile of positive SHAP values for feature j.

A buying window is flagged when $\text{CS}_i \geq 3$, indicating that multiple signals are piling up simultaneously. This provides sales teams with actionable, interpretable recommendations: "Contact customer i because features A, B, and C are all elevated."

3.5 Survival Data Construction

For each customer, survival pairs (T_i, δ_i) are constructed where T_i represents the days since the last order (capped at the censoring threshold) and $\delta_i = 1$ if the customer ordered within 180 days, 0 otherwise. Customers who have not ordered within this threshold are considered right-censored, as it cannot be determined whether they will return.

3.6 Model Validation

Model performance is evaluated using the concordance index (C-index) [27], the standard metric for survival model discrimination. The C-index measures the probability that, for a randomly selected pair of customers, the model correctly ranks their relative risk:

$$C = \frac{\sum_{i,j} \mathbf{1}[T_i < T_j] \cdot \mathbf{1}[\hat{r}_i > \hat{r}_j] \cdot \delta_i}{\sum_{i,j} \mathbf{1}[T_i < T_j] \cdot \delta_i} \tag{6}$$

where $\hat{r}_i$ is the predicted risk score for customer i. A C-index of 0.5 indicates random prediction, while 1.0 indicates perfect discrimination.

A two-level splitting strategy is employed. First, the customers are split 80/20 into a training set and a held-out test set, stratified by event status to maintain similar censoring proportions; this split is performed before any model fitting, and the test set is reserved exclusively for final performance estimation. Second, the training set is split 80/20 again, without stratification, to create an internal validation set used solely for early stopping, with training halting when the internal validation C-index fails to improve for 20 consecutive epochs.

To prevent information leakage, feature selection with RSF is performed on the training set only, using the same split, so that the held-out test data influences neither feature selection nor model fitting. The reported test C-index is therefore an estimate on customers never seen during any stage of training.

4 Dataset

The framework is evaluated on transaction data from a global industrial equipment manufacturer specializing in spare-part sales. The dataset captures B2B sales transactions across multiple European and international markets.

4.1 Data Description

The dataset comprises 17,261 transaction records spanning from January 2022 to November 2025, representing approximately four years of sales activity (Table 1). Each transaction record contains order-level information, including customer identifier, order date, delivery date, item details, quantities, and revenue. It includes both sales orders and quotations (offers). For survival analysis, the focus is on completed transactions with status 'Invoiced', 'SOLD', or 'Delivered', yielding 13,563 confirmed sales records across 779 unique customers. Table 2 presents the distribution of the order statuses in the data set.

Table 1 Dataset Overview

Characteristic	**Value**
Total transaction records	17,261
Observation period	Jan 2022 – Nov 2025
Duration (days)	1,411
Unique customers	870
Unique items (SKUs)	3,229
Unique target devices	1,041
Countries represented	44

The customer base is predominantly European, with Germany and Italy representing the largest markets. Table 3 shows the top ten countries by transaction volume.

Table 2 Distribution of Order Status

Order Status	Count
Invoiced	10,242
SOLD	3,227
LOST	1,891
READY	747
Active	471
Cancelled	238
Other	445

Table 3 Geographic Distribution of Transactions (Top 10 Countries)

Country	Transactions	% of Total
Germany	6,588	38.2%
Italy	3,175	18.4%
Switzerland	1,066	6.2%
France	1,006	5.8%
Austria	933	5.4%
United Kingdom	798	4.6%
Belgium	484	2.8%
Netherlands	421	2.4%
United States	400	2.3%
Sweden	344	2.0%

4.2 Customer Purchase Behavior

For the buying-window analysis, the focus is on repeat customers—those with two or more completed orders—since these provide the inter-order timing signal required for survival modeling. After applying all preprocessing exclusions, 523 repeat customers remain in the analysis dataset. Each customer appears exactly once in the survival dataset (one row per customer), ensuring that stratified cross-validation produces non-overlapping train-validation splits without customer-level data leakage.

Table 4 presents descriptive statistics for repeat customers.

The median customer places 4 orders over an active span of approximately 1.5 years (535 days), with a typical inter-order gap of 112 days. Substantial heterogeneity

Table 4 Repeat Customer Characteristics (n = 523)

Variable	Mean	Std	Median	Min	Max
Number of orders	6.1	5.9	4.0	2	40
Total revenue (EUR)	16,666	25,607	7,933	126	189,464
Mean inter-order gap (days)	162	171	112	1	1,052
Active span (days)	557	414	535	1	1,389

exists in purchasing patterns, with inter-order gaps ranging from 1 day to nearly 3 years.

4.3 Data Preprocessing

Prior to feature engineering, the following preprocessing steps were applied:

1. **Date parsing**: Order and delivery dates were converted from integer format (YYYYMMDD) to datetime objects.
2. **Currency normalization**: Swedish orders (identified by order type "Order (SE)" or "Offer (SE)") were converted from SEK to EUR using the exchange rate 1 EUR = 11 SEK.
3. **Product exclusions**: Upgrade products and 20 non-spare-part product classes were excluded to focus on regular maintenance purchases.
4. **Status filtering**: Transactions were filtered to completed orders (status: Invoiced, SOLD, or Delivered) to ensure clean survival signals.
5. **Customer aggregation**: Transaction-level records were aggregated to customer-level features including order frequency, revenue patterns, item diversity, and temporal characteristics.
6. **Repeat customer filtering**: For the primary analysis, customers with at least two orders were retained to enable inter-order timing analysis.

After preprocessing, the analysis dataset contains 12,970 transactions from 523 repeat customers.

5 Feature Engineering

A total of 21 customer-level features were engineered from the transaction data, organized into four categories: behavioral features capturing purchasing patterns, financial features reflecting order value, temporal features describing customer tenure, and inter-order gap features characterizing purchase rhythm. Table 5 presents the complete feature set.

Table 5 Engineered Customer-Level Features

Feature	Category	Description
num_orders	Behavioral	Total number of distinct orders
num_order_lines	Behavioral	Total transaction line items
total_quantity	Behavioral	Cumulative quantity ordered
avg_quantity_per_order	Behavioral	Mean quantity per transaction
max_quantity_single_order	Behavioral	Largest single order quantity
unique_items	Behavioral	Number of distinct SKUs purchased
unique_devices	Behavioral	Number of distinct target devices served
items_per_device	Behavioral	SKU diversity per device
item_diversity_ratio	Behavioral	Unique items / total order lines
total_revenue	Financial	Cumulative revenue (EUR)
avg_order_value	Financial	Mean order value
max_order_value	Financial	Largest single order value
revenue_per_order	Financial	Total revenue / number of orders
active_span_days	Temporal	Days between first and last order
orders_per_year	Temporal	Annualized order frequency
mean_gap	Inter-order	Average days between consecutive orders
std_gap	Inter-order	Standard deviation of inter-order gaps
min_gap	Inter-order	Shortest gap between orders
max_gap	Inter-order	Longest gap between orders
last_gap	Inter-order	Most recent inter-order gap
gap_cv	Inter-order	Coefficient of variation (std/mean)

The inter-order gap features are particularly relevant for buying window detection, as they capture each customer's purchasing rhythm. The coefficient of variation (`gap_cv`) distinguishes regular purchasers (low CV) from irregular ones (high CV), informing whether cycle-based timing predictions are reliable for a given customer.

6 Experimental Setup

Implementation. The framework is implemented in Python using scikit-survival for Random Survival Forest and PyTorch for DeepSurv. SHAP values are computed using the KernelExplainer from the SHAP library [8], which provides model-agnostic explanations compatible with neural network architectures containing batch normalization.

Random Survival Forest. The RSF model uses 500 trees with a minimum of 10 samples per leaf node. Feature importance is computed via permutation importance with 10 repetitions per feature to estimate variance. The elbow detection algorithm locates the largest relative drop between consecutive values in the ranked importance scores, subject to a floor of five retained features.

DeepSurv. Table 6 summarizes the network architecture and training configuration. The two-layer architecture with decreasing width (64→32) was selected to balance model capacity with the modest sample size ($n = 523$). Dropout and batch normalization provide regularization, while early stopping prevents overfitting on the validation set.

Table 6 DeepSurv Network Architecture and Training Configuration

Category	Component	Specification
Architecture	Input layer	7 features (4 profile + 3 purchase cycle)
	Hidden layer 1	64 units, ReLU, BatchNorm, Dropout(0.3)
	Hidden layer 2	32 units, ReLU, BatchNorm, Dropout(0.3)
	Output layer	1 unit (log-hazard ratio)
Training	Optimizer	Adam (lr = 10^{-3}, weight decay = 10^{-4})
	Batch size	64
	Maximum epochs	500
	Early stopping	Patience of 20 epochs
	Data split	Two-level 80/20: train/test stratified by event, then unstratified train/internal validation

SHAP. SHAP values are computed using KernelExplainer with 100 background samples drawn at random from the full set of customers being explained. For each

customer, 100 SHAP samples are computed to estimate feature contributions. The "elevated" threshold for each feature is set at the 75th percentile of positive SHAP values, ensuring that only the top quartile of contributions are flagged.

Evaluation. Model performance is evaluated using three metrics: the concordance index (C-index), which measures the probability of correctly ranking customer pairs by risk [27]; the convergence score distribution, capturing the frequency of customers at each CS level; and the buying window detection rate, defined as the percentage of customers flagged with CS ≥ 3.

7 Results

7.1 Selected Features

To prevent information leakage from validation data into feature selection, an 80/20 stratified train-holdout split was applied before training RSF. A Random Survival Forest with 500 trees was trained on the training set (418 customers) using all 21 engineered features. The model achieved a concordance index of 0.861 on training data and 0.688 on the holdout set, indicating good generalization.

Table 7 presents the top features ranked by permutation importance computed on training data only. The largest relative drop in importance occurs immediately after `active_span_days`, so the elbow criterion alone would retain a single feature; the floor of five retained features is therefore binding, and five features were carried forward to the DeepSurv model. This is consistent with the ablation analysis reported in Section 7, in which tenure accounts for the largest share of predictive performance.

The dominant feature is `active_span_days` (importance = 0.0745), substantially higher than the remaining features. Notably, `last_gap`—a purchase cycle feature—emerges as the second most important feature, indicating that recent inter-order timing

Table 7 Feature Importance from Random Survival Forest (Train C-index = 0.861, Holdout C-index = 0.688)

Feature	Importance	Std
active_span_days	0.0745	0.009
last_gap	0.0154	0.002
max_order_value	0.0139	0.002
avg_order_value	0.0112	0.002
revenue_per_order	0.0112	0.001
Selected features: top 5 (minimum-feature floor binding)		

contributes predictive value. The feature set is augmented with two additional purchase cycle features (`mean_gap`, `gap_cv`) to operationalize the buying window concept. The final seven features are organized into three categories:

- **Tenure** (1 feature): `active_span_days`
- **Financial** (3 features): `max_order_value`, `avg_order_value`, `revenue_per_order`
- **Purchase cycle** (3 features): `last_gap`, `mean_gap`, `gap_cv`

7.2 DeepSurv Model

A DeepSurv neural network was trained using the seven features selected by RSF, applying the same 80/20 stratified split to ensure the test set (105 customers) remains fully held out. The training set (418 customers) was further split internally (80/20) for early stopping during training. Table 8 presents the training results.

Table 8 DeepSurv Model Performance

Metric	Value
Training samples	418
Held-out test samples	105
Features	7 (1 tenure + 3 financial + 3 purchase cycle)
Epochs trained	31
Held-out test C-index	0.735

Training ran for 31 epochs, with best internal validation performance at epoch 11 and early stopping triggered after 20 epochs without improvement. The held-out test concordance index of 0.735 indicates meaningful discriminative ability, representing strong predictive performance above random chance (C-index = 0.5).

Figure 1 shows the training progression. The left panel displays the Cox partial likelihood loss decreasing over epochs for both training and internal validation sets. The right panel shows the concordance index stabilizing around 0.73. The close tracking of training and validation curves indicates that the model generalizes appropriately without overfitting.

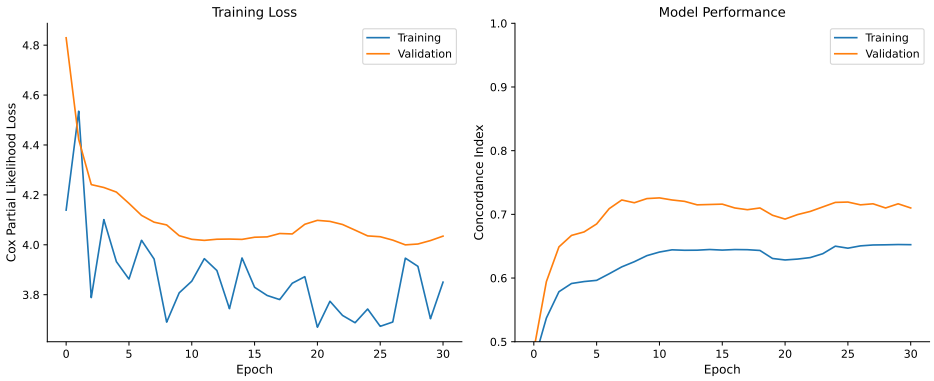


Fig. 1 DeepSurv training curves. Left: Cox partial likelihood loss over epochs. Right: Concordance index over epochs. Training and validation curves track closely, indicating no overfitting

7.2.1 Model Validation and Generalization

The final validation on the fully held-out test set produced a concordance index of 0.735, indicating that the model generalizes with meaningful predictive performance to previously unseen customers.

7.2.2 Robustness and Baseline Comparison

In addition to the model performance reported based on the held-out test set, robustness and diagnostic analyses were conducted using the full dataset. These analyses

are not intended to assess generalization performance, but rather to examine the stability and behavior of the model under varying conditions, such as different random initializations and label perturbations.

Multi-seed stability. The DeepSurv model was trained with five different random seeds (42, 123, 456, 789, 1024), using consistent hyperparameters. Each seed reseeds both the network initialization and the internal validation partition. The model achieved a mean C-index of 0.684 ± 0.046 across seeds, indicating stable performance across these sources of randomness.

Permutation test. To confirm that the model learns genuine signal rather than spurious patterns, DeepSurv was trained on permuted survival labels. The permuted model achieved a C-index of 0.557, near the expected value of 0.5 for random prediction. This confirms that the model performance on real data reflects learned structure rather than overfitting to noise.

Cross-validation comparison. Using 5-fold stratified cross-validation, DeepSurv was compared against Cox Proportional Hazards [14] and Random Survival Forest. Table 9 presents the results.

Table 9 Model Comparison (5-fold Stratified Cross-Validation)

Model	Mean C-index	Std
DeepSurv	0.685	0.043
Cox Proportional Hazards	0.683	0.053
Random Survival Forest	0.673	0.043

All three models achieve comparable cross-validated performance ($C \approx 0.67$–0.69), with DeepSurv and Cox PH showing nearly identical discrimination.

Kaplan-Meier stratification. To validate that predicted risk scores correspond to actual survival differences, customers were stratified into terciles based on DeepSurv

risk scores and computed Kaplan-Meier survival curves [28]. Table 10 presents the event rates by risk group.

Table 10 Kaplan-Meier Stratification by Predicted Risk Terciles

Risk Group	Customers	Events	Event Rate
Low Risk	174	29	16.7%
Medium Risk	175	61	34.9%
High Risk	174	77	44.3%

The log-rank test [29] comparing high- versus low-risk groups yields $p < 10^{-7}$. The event rate increases monotonically from low-risk (16.7%) to high-risk (44.3%) groups. Figure 2 visualizes the Kaplan-Meier survival curves for each risk group.

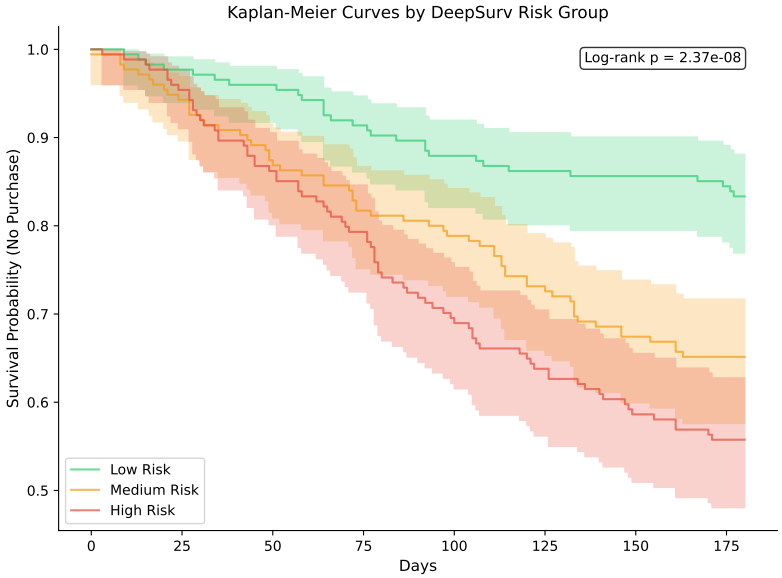


Fig. 2 Kaplan-Meier survival curves stratified by DeepSurv predicted risk terciles. High-risk customers (red) exhibit rapid purchasing behavior, while low-risk customers (green) show minimal activity. Shaded regions indicate 95% confidence intervals

7.2.3 Feature Ablation Study

To understand the contribution of different feature groups, an ablation study was conducted, systematically removing or isolating feature categories. Table 11 presents the results using 5-fold cross-validation.

Table 11 Feature Ablation Study (5-fold Cross-Validation)

Configuration	Features	C-index	Δ vs Full
Full Model	7	0.706 ± 0.034	—
Removal experiments			
No Purchase Cycle	4	0.687 ± 0.038	-2.7%
No Financial	4	0.690 ± 0.036	-2.3%
No Tenure	6	0.615 ± 0.034	-12.9%
Isolation experiments			
Only Purchase Cycle	3	0.557 ± 0.051	-21.2%
Only Financial	3	0.625 ± 0.045	-11.5%

The seven features are organized into three semantic categories: *tenure* (`active_span_days`), *financial* (`max_order_value`, `revenue_per_order`, `avg_order_value`), and *purchase cycle* (`mean_gap`, `last_gap`, `gap_cv`). Removing tenure causes the largest performance drop (12.9%), confirming that customer longevity is the strongest predictor. Removing purchase cycle features reduces performance by 2.7%, indicating that timing features contribute meaningful signal. Purchase cycle features alone achieve C-index = 0.557.

7.3 Buying Window Detection

SHAP decomposition was applied to the trained DeepSurv model to identify customers whose buying windows may be opening. For each customer, the contribution of each feature to their predicted hazard was computed. A feature was considered "elevated" if its SHAP value exceeded the 75th percentile of positive SHAP values for that feature. This threshold is a design parameter: the 75th percentile identifies customers in the

upper quartile of hazard contribution for each feature, providing a principled default that can be adjusted based on organizational sales capacity or, ideally, calibrated against conversion data in deployment.

Table 12 presents the per-feature thresholds for elevated status.

Table 12 SHAP Thresholds for Elevated Feature Status

Feature	Category	Threshold
max_order_value	Financial	0.358
active_span_days	Tenure	0.311
revenue_per_order	Financial	0.076
last_gap	Purchase Cycle	0.060
avg_order_value	Financial	0.051
mean_gap	Purchase Cycle	0.011
gap_cv	Purchase Cycle	0.001

To validate that SHAP-based feature importance generalizes to unseen customers, importance was computed separately on training (n=418) and test (n=105) data. Table 13 presents the comparison.

Table 13 SHAP Importance Generalization: Train vs Test Data

Feature	Train	Test	Train Rank	Test Rank
max_order_value	0.193	0.144	1	2
revenue_per_order	0.173	0.085	2	3
active_span_days	0.164	0.177	3	1
avg_order_value	0.082	0.045	4	4
last_gap	0.032	0.029	5	5
mean_gap	0.008	0.008	6	6
gap_cv	0.001	0.001	7	7

Notes: Importance = mean |SHAP|. Correlation between train and test importance: $r = 0.90$.

The feature importance ranking is highly consistent between train and test data, with a correlation of 0.90. Minor rank differences in the top three features reflect

natural sampling variability, while the bottom four features maintain identical ordering. This indicates that the feature importance structure underlying the framework is stable across partitions. It does not by itself establish that the flagging decision generalizes, which would require evaluation against observed conversion outcomes.

The convergence score counts how many features are simultaneously elevated for each customer. Table 14 shows the distribution of convergence scores.

Table 14 Distribution of Convergence Scores

Convergence Score	Customers	Percentage
0	256	48.9%
1	140	26.8%
2	65	12.4%
3	44	8.4%
4	15	2.9%
5	3	0.6%

Using a threshold of CS ≥ 3, 62 customers (11.9%) were identified whose buying windows may be opening. The convergence threshold of 3 is a design parameter: with 7 features, requiring at least three to be simultaneously elevated ensures that flagged customers exhibit genuine multi-dimensional signal convergence rather than single-feature spikes. These customers exhibit simultaneous elevation across multiple feature categories, indicating convergence of purchase signals. The SHAP decomposition provides interpretable explanations for each flagged customer, identifying which specific features are driving the elevated hazard prediction.

7.3.1 Threshold Sensitivity Analysis

To examine how threshold choices affect the flagging rate, a sensitivity analysis was conducted varying both the SHAP percentile threshold (70th, 75th, 80th) and the convergence score threshold (CS $\geq 2, 3, 4$). Table 15 presents the results.

Table 15 Sensitivity Analysis: Flagging Rate (%) by Threshold Parameters

CS Threshold	70th pct	75th pct	80th pct
CS ≥ 2	29.3%	24.3%	20.1%
CS ≥ 3	15.7%	**11.9%**	7.1%
CS ≥ 4	4.6%	3.4%	2.5%

Notes: Bold indicates baseline configuration used in this study.

The flagging rate ranges from 2.5% (strict: 80th percentile, CS $\geq$ 4) to 29.3% (lenient: 70th percentile, CS $\geq$ 2). The baseline configuration (75th percentile, CS $\geq$ 3) represents a moderate setting that flags approximately one in eight customers. Organizations can adjust these parameters based on sales team capacity: a smaller team might prefer stricter thresholds (fewer, higher-confidence flags), while a larger team could use more lenient settings for broader coverage. Ideally, these thresholds should be calibrated against actual conversion data in deployment.

8 Discussion

8.1 Interpretation of Results

Predictive performance. The DeepSurv model achieves a held-out test C-index of 0.735, meaning that for 73.5% of comparable customer pairs, the model correctly identifies which customer will reorder sooner. This performance is meaningful for B2B sales prediction where customer behavior is influenced by numerous external factors beyond the available features.

Model comparison. All three survival models—DeepSurv, Cox Proportional Hazards, and Random Survival Forest—achieve comparable performance ($C \approx 0.67$–0.70). The modest improvement of DeepSurv over Cox PH suggests that the survival relationship is largely captured by linear effects, with minor gains from non-linear interactions. All three models are compatible with SHAP-based explanation: Cox

PH via KernelExplainer, RSF via TreeExplainer, and DeepSurv via gradient-based methods. DeepSurv is retained because it demonstrates lower variance across folds, the proportional hazards structure ensures additive SHAP decomposition, and neural network flexibility allows capturing non-linear patterns should they emerge in other datasets.

Feature contributions. The ablation study reveals that customer tenure (`active_span_days`) is the most predictive feature, with removal causing a 12.9% performance drop. Longer-tenured customers exhibit more predictable purchasing patterns, making their reorder timing easier to forecast. Removing purchase cycle features reduces performance by 2.7%, indicating that timing features—particularly `last_gap`, which emerged as the second most important feature in RSF selection—contribute meaningful predictive signal. Purchase cycle features alone achieve C-index = 0.557, substantially above random prediction but insufficient without profile features.

Rationale for retaining full feature set. While tenure dominates prediction, the full seven-feature set is retained because each category provides distinct interpretive value. Profile features (tenure, financial) capture customer characteristics—who they are—while purchase cycle features capture historical purchasing rhythm—when they typically buy. Notably, `last_gap` emerged as naturally selected by RSF (second highest importance), validating the inclusion of timing features without manual augmentation. A customer flagged due to high order value history warrants different engagement than one flagged due to long tenure: the former may be receptive to premium offerings, while the latter may simply be due for routine reorder. The SHAP decomposition identifies which specific factors drive each customer's elevated hazard, enabling tailored sales engagement.

SHAP thresholds and convergence. The SHAP analysis reveals that `max_order_value` has the highest threshold for elevated status (0.358), followed by `active_span_days` (0.311), indicating that financial and tenure features contribute

most strongly to elevated hazard predictions. The distribution of convergence scores shows that most customers (88.1%) have fewer than three elevated features at any given time, while 11.9% exhibit the signal convergence characteristic of an opening buying window. This suggests the framework successfully distinguishes between routine customer states and moments of heightened purchase receptivity. Importantly, the SHAP importance ranking remains stable on held-out test data ($r = 0.90$), with minor rank differences in the top features reflecting natural sampling variability, indicating that the feature structure the framework relies on is not an artefact of the training partition.

8.2 Managerial Implications

The framework provides several practical benefits for B2B sales operations:

Prioritized contact lists. The convergence score enables generation of weekly priority reports ranking customers by urgency of contact. Sales teams receive actionable lists identifying not just *whom* to contact, but *why* — with specific features driving each recommendation. A sample output might indicate: "Contact Customer X: 4 signals elevated (Active Span, Max Order Value, Mean Gap, Last Gap); 85% through typical purchase cycle."

Resource allocation efficiency. Rather than contacting all customers on fixed schedules, sales resources can be directed toward the subset with opening buying windows, improving intervention scheduling efficiency. As Table 15 demonstrates, the prioritization policy can be tuned from 2.5% to 29.3% flagging rate depending on sales team capacity and risk tolerance. The baseline configuration flags approximately 12% of customers, but organizations with smaller teams might prefer stricter thresholds for fewer, higher-confidence recommendations.

Interpretable recommendations. Unlike black-box propensity scores, the SHAP decomposition provides transparency. Sales representatives understand *why* a

customer is flagged, enabling more informed and personalized conversations. A customer flagged due to timing features (approaching typical reorder interval) warrants a different approach than one flagged due to behavioral features (historically high order values).

Intervention timing optimization. The gap progress metric (days since last order relative to typical gap) provides explicit timing guidance. Customers at 70–100% of their typical purchase cycle represent the optimal intervention window — early enough to influence the decision, but close enough that the need is becoming salient.

Differential timing opportunities. Figure 3 reveals that buying windows open at different stages of the purchase cycle depending on which signals converge first. This creates distinct types of sales opportunities. *Early detection* (0–50% of typical cycle) occurs when strong behavioral signals trigger detection before the customer reaches their typical reorder timing. These customers may be experiencing accelerated equipment wear or expanding operations; early contact provides opportunity for cross-selling complementary spare parts, building relationship momentum, and consultative engagement before urgent need narrows the customer's focus. *On-cycle detection* (70–120% of typical cycle) occurs when both behavioral and timing signals converge near the expected reorder point, indicating standard reorder facilitation is appropriate. *Late detection* (>120% of typical cycle) occurs when timing features dominate, indicating an overdue customer; contact is urgent but the customer may have already purchased elsewhere or may require immediate fulfillment rather than consultative selling. The framework thus provides not only *when* to contact, but guidance on *how* to approach the conversation based on the composition of elevated signals.

Notably, early detection is not purely behavioral. Figure 3 reveals that some customers flagged early in their cycle (0–50% gap progress) exhibit elevated timing features rather than behavioral ones. This occurs when customers have historically irregular purchasing patterns — high gap coefficient of variation (`gap_cv`) or unusual

historical gaps (`mean_gap`, `last_gap`). Such customers do not follow predictable cycles, making them worth contacting even before their "typical" reorder time. This observation strengthens the framework's value proposition: the model captures both customer propensity and timing unpredictability, rather than simply recommending contact at 100% of the typical cycle.

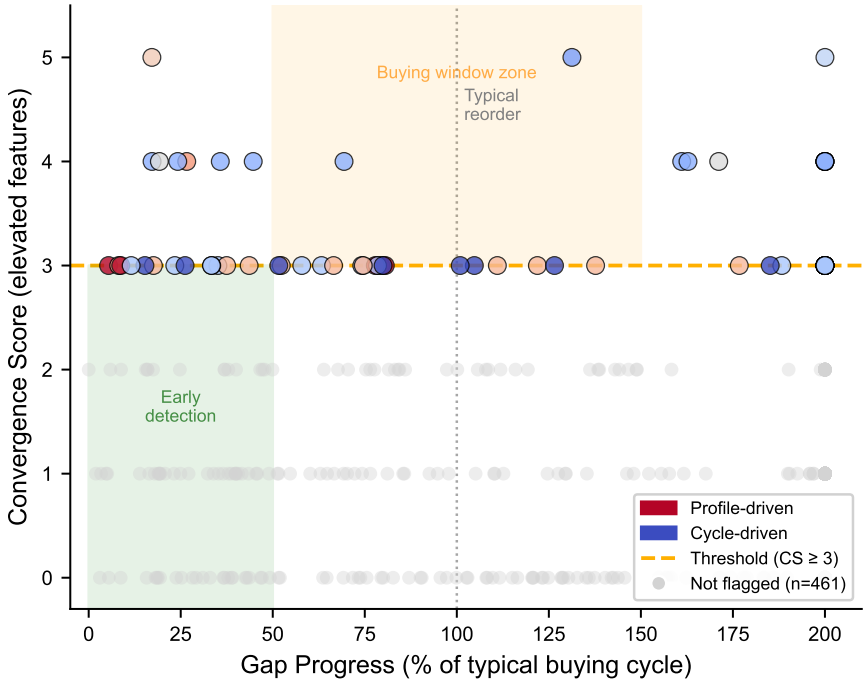


Fig. 3 Signal buildup across the buying cycle. Each point represents a customer; colored points indicate those with convergence score ≥ 3 (buying window opening). Point color reflects whether behavioral (red) or timing (blue) features dominate the elevated signal set. The framework detects buying windows at different cycle stages: early detection when behavioral signals are strong, on-cycle detection when both signal types converge, and late detection for overdue customers where timing signals dominate

8.3 Limitations

Several limitations should be acknowledged. First, the framework requires customers with purchase history (at least two orders) to compute inter-order gap features, limiting applicability to repeat customers. Extension to new customer acquisition would require alternative feature engineering approaches.

Second, the 75th percentile threshold for elevated SHAP status and the convergence threshold of CS $\geq$ 3 were chosen based on interpretability rather than optimized for business outcomes. The sensitivity analysis (Table 15) demonstrates how flagging rates vary across parameter combinations, but future work should validate optimal thresholds against actual sales conversion data.

Third, the elevated thresholds are percentiles of the SHAP values computed across the cohort being scored, so a customer's flagged status is defined relative to that cohort rather than by an independent per-customer criterion. The detection stage therefore produces a prioritized ranking within a scored population; applying it to a newly acquired customer requires recomputing the thresholds against a comparable cohort rather than applying a fixed rule.

Fourth, the current analysis uses a static snapshot of customer features. In practice, the model should be re-run periodically (e.g., weekly) to capture evolving customer states and generate updated priority lists.

Finally, the framework assumes that purchase timing is primarily customer-driven. In contexts where supplier-initiated promotions or external factors strongly influence timing, additional features capturing these dynamics may be necessary.

9 Conclusion

This paper introduced a survival-based framework for identifying optimal sales intervention timing in B2B industrial spare-part sales. The key contribution is the concept

of SHAP-based convergence detection—identifying when multiple hazard-driving signals simultaneously exceed their respective thresholds, indicating that a customer's buying window may be opening.

The three-stage framework combines Random Survival Forest for feature selection, DeepSurv for survival prediction, and SHAP decomposition for interpretable buying window detection. Validation on industrial transaction data (523 repeat customers, 12,970 transactions) demonstrated meaningful predictive performance (C-index = 0.735 on held-out test data) and identified 62 customers (11.9%) with opening buying windows. Unlike traditional propensity scores, the convergence score provides transparency into *why* each customer is flagged, enabling targeted sales engagement. The framework adapts buying window timing to each customer's unique purchase pattern, with timing thresholds personalizing to individual cycles rather than applying universal rules.

Importantly, the convergence detection mechanism is model-agnostic. Since Cox Proportional Hazards, Random Survival Forest, and DeepSurv achieved comparable performance in our evaluation ($C \approx 0.67$–0.69), organizations may select the survival model that best fits their technical infrastructure. Cox PH offers simplicity and widespread availability in statistical software; DeepSurv provides flexibility for capturing non-linear effects should they emerge in other datasets. The SHAP-based convergence detection can be applied to any model producing continuous risk scores.

Future work should validate the framework against actual sales conversion outcomes through controlled deployment, explore extension to new customer acquisition where purchase history is unavailable, and investigate streaming updates for continuous monitoring as new transactions arrive. The convergence detection approach may generalize to other B2B sales contexts beyond industrial spare parts—such as professional services, SaaS subscription renewals, or consulting engagements—where

identifying the optimal moment to contact buyers before their buying windows open can improve both sales effectiveness and customer experience.

Acknowledgments

The authors thank participating sales organizations for providing authentic B2B sales data. This research was part of the PATA co-research project (AI-enabled customer experiences), in cooperation with the University of Turku, Finland, the Haaga-Helia University of Applied Sciences, Finland.

Author Contributions

Joona Mäntyvaara: Conceptualization, Writing - Original Draft, Data curation, Formal analysis, Investigation, Methodology, Software, Validation, Visualization, Funding acquisition. **Tanja Vähämäki**: Conceptualization, Writing - Review & Editing, Data curation, Validation. **Paavo Nevalainen**: Writing - Review & Editing. **Jukka Heikkonen**: Writing - Review & Editing, Funding acquisition, Project administration, Supervision.

Declarations

Conflict of Interest

The authors declare no conflict of interest. This research was financially supported by a Business Finland Co-research project. The funder had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Funding

Business Finland supported this co-research project. Funding number: 6684/31/2023. The funding institution had no role in study design, data collection and analysis, decision to publish, or manuscript preparation.

Data Availability Statement

The data that support the findings of this study are available from the corresponding author upon reasonable request. Due to privacy and confidentiality agreements with the partner company, the raw data cannot be made publicly available.

Ethics, Consent to Participate, and Consent to Publish declarations

Not applicable. This study did not involve human participants or animals, and no participants were recruited. The analysis is based on organisation-level business transaction records provided by the partner company; no personal data concerning identifiable individuals were collected or processed.

References

[1] Rackham N. SPIN Selling. McGraw-Hill; 1988.

[2] Anderson JC, Narus JA. Business marketing: understand what customers value. Harvard Business Review. 1998;.

[3] Paschen J, Wilson M, Ferreira JJ. Collaborative intelligence: How human and artificial intelligence create value along the B2B sales funnel. Business Horizons. 2020;63(3):403–414.

[4] Moncrief WC, Marshall GW. The evolution of the seven steps of selling. Industrial Marketing Management. 2005;34(1):13–22.

[5] Reinartz W, Kumar V. The impact of customer relationship characteristics on profitable lifetime duration. Journal of Marketing. 2003;67(1):77–99.

[6] Ishwaran H, Kogalur UB, Blackstone EH, Lauer MS. Random survival forests. The Annals of Applied Statistics. 2008;2(3):841–860.

[7] Katzman JL, Shaham U, Cloninger A, Bates J, Jiang T, Kluger Y. DeepSurv: personalized treatment recommender system using a Cox proportional hazards deep neural network. BMC Medical Research Methodology. 2018;18(1):24.

[8] Lundberg SM, Lee SI. A unified approach to interpreting model predictions. In: Advances in Neural Information Processing Systems. vol. 30; 2017. .

[9] Schmittlein DC, Morrison DG, Colombo R. Counting your customers: Who are they and what will they do next? Management Science. 1987;33(1):1–24.

[10] Fader PS, Hardie BG. A note on deriving the Pareto/NBD model and related expressions. Marketing Science. 2005;.

[11] Gupta S, Hanssens D, Hardie B, Kahn W, Kumar V, Lin N, et al. Modeling customer lifetime value. Journal of Service Research. 2006;9(2):139–155.

[12] Tamaddoni Jahromi A, Stakhovych S, Ewing M. Managing B2B customer churn, retention and profitability. Industrial Marketing Management. 2014;43(7):1258–1268.

[13] Sanchez Ramirez J, Coussement K, De Caigny A, Benoit DF, Guliyev E. Incorporating usage data for B2B churn prediction modeling. Industrial Marketing Management. 2024;120:191–205.

[14] Cox DR. Regression models and life-tables. Journal of the Royal Statistical Society: Series B (Methodological). 1972;34(2):187–220.

[15] Lee C, Zame W, Yoon J, Van Der Schaar M. DeepHit: A deep learning approach to survival analysis with competing risks. Proceedings of the AAAI Conference on Artificial Intelligence. 2018;32(1).

[16] Kvamme H, Borgan Ø, Scheel I. Time-to-event prediction with neural networks and Cox regression. Journal of Machine Learning Research. 2019;20(129):1–30.

[17] Wang P, Li Y, Reddy CK. Machine learning for survival analysis: A survey. ACM Computing Surveys. 2019;51(6):1–36.

[18] Lipton ZC. The mythos of model interpretability. Queue. 2018;16(3).

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

[20] Hallikainen H, Savimäki E, Laukkanen T. Fostering B2B sales with customer big data analytics. Industrial Marketing Management. 2020;86:90–98.

[21] Ribeiro MT, Singh S, Guestrin C. "Why should I trust you?": Explaining the predictions of any classifier. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2016;p. 1135–1144.

[22] Molnar C, Casalicchio G, Bischl B. Interpretable machine learning–A brief history, state-of-the-art and challenges. ECML PKDD 2020 Workshops. 2020;p. 417–431.

[23] Chen V, Li Q, Zhu KQ, et al. Explaining models: An empirical study of how explanations impact fairness judgment. In: Proceedings of the 2020 Conference

on Fairness, Accountability, and Transparency; 2020. .

[24] Ioffe S, Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: Proceedings of the 32nd International Conference on Machine Learning (ICML); 2015. p. 448–456.

[25] Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R. Dropout: A simple way to prevent neural networks from overfitting. Journal of Machine Learning Research. 2014;15(1):1929–1958.

[26] Kingma DP, Ba J. Adam: A method for stochastic optimization. In: Proceedings of the 3rd International Conference on Learning Representations (ICLR); 2015. .

[27] Harrell FE, Califf RM, Pryor DB, Lee KL, Rosati RA. Evaluating the yield of medical tests. JAMA. 1982;247(18):2543–2546.

[28] Kaplan EL, Meier P. Nonparametric estimation from incomplete observations. Journal of the American Statistical Association. 1958;53(282):457–481.

[29] Mantel N. Evaluation of survival data and two new rank order statistics arising in its consideration. Cancer Chemotherapy Reports. 1966;50(3):163–170.

Appendix A Sample Priority Report

Table A1 presents an excerpt from a weekly priority report generated by the framework. This report format is designed for sales team use, providing actionable contact recommendations with interpretable explanations.

The report demonstrates several key aspects of the framework. **Heterogeneous timing**: Customers range from 17% to 711% through their typical purchase cycles, yet all are flagged due to signal convergence; Customer C010 (17% progress) is flagged

Table A1 Sample Monday Morning Priority Report (Top 10 Customers)

Rank	Customer	CS	Gap	Progress	Why Contact
1	C001	5	60	129%	Active Span, Max Order Value, Avg Order Value, Revenue Per Order, Mean Gap
2	C002	5	31	711%	Active Span, Max Order Value, Revenue Per Order, Mean Gap, Gap CV
3	C003	3	71	78%	Active Span, Max Order Value, Avg Order Value
4	C004	3	63	103%	Active Span, Max Order Value, Revenue Per Order
5	C005	3	126	78%	Active Span, Avg Order Value, Revenue Per Order
6	C006	3	45	72%	Active Span, Avg Order Value, Gap CV
7	C007	4	55	200%	Active Span, Max Order Value, Avg Order Value, Mean Gap
8	C008	4	43	159%	Active Span, Max Order Value, Avg Order Value, Mean Gap
9	C009	4	86	23%	Active Span, Max Order Value, Revenue Per Order, Mean Gap
10	C010	5	734	17%	Last Gap, Avg Order Value, Revenue Per Order, Mean Gap, Gap CV

Notes: CS = Convergence Score (number of elevated features). Gap = typical inter-order gap in days. Progress = days since last order as percentage of typical gap. Why Contact = features exceeding their SHAP thresholds.

early due to strong profile signals indicating a high-value, long-tenured customer, while C002 (711% progress) is significantly overdue and exhibits urgent signal convergence. **Interpretable recommendations**: Each customer receives specific reasons for contact; a sales representative approaching Customer C007 knows that four signals are elevated (Active Span, Max Order Value, Avg Order Value, Mean Gap), enabling a more informed conversation. **Actionable prioritization**: The convergence score provides clear ranking—customers with five elevated features (C001, C002, C010) should be contacted first, as they exhibit the strongest buying window signals.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- SupplementWhentoContacttheBuyerAnExplainableSurvivalFrameworkforSalesInterventionPrioritizationSpringer.zip