SOCIAL AND MANAGEMENT SCIENCES RESEARCH JOURNAL

www.smsrj.com

**p-ISSN:** 2791-0415 **e-ISSN:** 2791-0423

**Volume:4 Issue: 02(2026)**

# Adaptive Risk-Aware Decision-Focused Learning for Dynamic Inventory Optimization with Lead Times and Censored Demand

***Jasper Moore [1], Mia Edwards [2], Katherine Kang [3]***

*[1]Graduate School of Management, University of California, Davis, USA*

*[2]Poole College of Management, North Carolina State University, USA*

*[3]College of Business, Florida State University, USA*

***Corresponding author:*** *Jasper Moore*

***Email:*** *j19.moore@ucdavis.edu*

**Abstract:**

*Decision-focused learning trains a demand forecaster on the cost of the decisions it induces, but in retail it has mostly been applied to single-period decisions, whereas replenishment is dynamic: stock carries over, orders arrive after a lead time and sales are censored when shelves empty. This paper proposes DynRA-DFL, an adaptive risk-aware decision-focused learning framework for periodic-review lost-sales systems. A global network emits monotone multi-horizon demand quantiles and a state-conditional risk weight; an order-up-to layer mixes the critical-fractile level with a closed-form CVaR-optimal level; and the network is trained through a differentiable simulator with deterministic or stochastic lead times, promotion-scaled penalties and a censoring-corrected mean–CVaR cost. An online risk calibration (ORC) rule updates a per-series safety margin from stock-out indicators and provably bounds the long-run stock-out frequency. On 909 M5 and 281 Stallion series, evaluated on observed sales and on reconstructed demand, risk-neutral dynamic training attains mean costs 11.2% lower than static single-period decision-focused learning and 5.9% lower than the same network used for predict-then-optimise, while adding CVaR only at the decision stage raises cost. DynRA-DFL trades off average inventory cost against selected tail-risk reductions: relative to the risk-neutral dynamic baseline it lowers the per-item tail cost by 10.4% at a 2.9% higher mean cost. Under a simulated demand surge, ORC raises the fill rate of the affected series for every policy, with cost changes ranging from −2.5% to +1.4%. We also quantify the bias of evaluating on censored sales.*

***Keywords***: *Index Terms—decision-focused learning, inventory control, lost sales, lead time, conditional value-at-risk, censored demand, online calibration..*

## 1. Introduction

Replenishment decisions in retail combine a demand forecast with an inventory policy. In most deployments the forecaster is fitted to a statistical criterion and its output is handed to the policy, a pipeline usually called predict-then-optimise (PTO). Large benchmarks have shown how much accuracy modern models can extract from retail data [1], but the cost of a replenishment decision is asymmetric and depends on what the system carries from one period to the next, so a more accurate forecast does not necessarily yield a cheaper policy. Decision-focused learning (DFL) addresses this mismatch by training the predictor on the loss of the decision it induces [2]–[5]. For inventory, DFL has mostly been instantiated as a single-period newsvendor in which a learner outputs a cost-optimal quantile directly [6], [7]. Zhang et al. [8] recently extended this line to a risk-aware multi-channel allocation layer with promotion-scaled shortage penalties and a censoring-consistent decision loss, and showed that attaching a CVaR objective to a predict-then-optimise pipeline raises cost whereas training through the risk-aware decision does not. Their model is single-period, and the authors identify multi-period dynamics with stochastic lead times as the natural next step. Multi-period end-to-end methods exist, but they either imitate dynamic-programming solutions computed in hindsight [9] or learn black-box ordering policies by reinforcement learning or differentiable simulation [10]–[13]; they are risk-neutral and do not expose a probabilistic forecast that planners can inspect.

Dynamics change the learning problem in three ways. A unit left over is not charged once but carried until it sells, and a lost sale is never recovered [14], [15], so the newsvendor fractile misprices both kinds of error. Risk accumulates over time: a planner is exposed to the bad weeks of individual items, not to an average over instances. Finally, sales are censored by availability [16], so a decision loss computed on observed sales rewards under-ordering, and demand drifts after training, so any offline calibration decays. This paper makes four contributions.

- *A differentiable risk-aware dynamic layer.* An order-up-to policy whose level mixes the critical fractile with a closed-form CVaR-optimal level (Proposition 1) is embedded in a lost-sales simulator with deterministic or stochastic lead times, and a multi-horizon quantile network is trained through it on a mean–CVaR objective of realised multi-period cost.
- *Censoring-aware dynamic training.* Holding costs are masked in censored periods and the carried-over state is imputed, which extends single-period censoring-consistent losses to carry-over dynamics.
- *Online risk calibration (ORC).* A one-line update of a per-series safety margin, driven only by stock-out indicators, with a distribution-free bound on the long-run stock-out frequency (Proposition 2).
- *Evidence on two real panels under two evaluation protocols.* DynRA-DFL trades average inventory cost against targeted tail-risk reductions, and the training-level risk preference selects a point on this frontier. The evaluation also yields design lessons: on M5 a single-period decision loss trails plain PTO, a constant risk weight suffices in place of a state-conditional one, and censored sales bias evaluation against censoring corrections.

## II. Related Work

### Decision-focused and end-to-end learning

SPO+ [2] and task-based learning [3] train predictors through the optimisation problem they feed, and recent surveys organise the field [4], [5]. Inventory applications learn newsvendor quantiles from features [6], [7], imitate hindsight-optimal multi-period decisions [9], or optimise neural policies by reinforcement learning [10], [11] and by pathwise gradients of simulated cost [12], [13]. Our layer keeps the forecast explicit: the network outputs anchored

quantiles, and decision-relevant adjustments pass through an interpretable order-up-to level and risk weight.

**Risk-averse and lost-sales inventory**

CVaR is a coherent risk measure with a convex representation [17]; CVaR newsvendor solutions [18] and risk-averse inventory models [19], [20] assume that the demand law is known. Nested risk measures give time-consistent dynamic programmes [21] but are hard to learn from data, so we use a static CVaR of period costs, which matches how planners judge bad weeks. Order-up-to policies are asymptotically optimal in lost-sales systems as the penalty-to-holding ratio grows [22], which motivates the policy class. Physics-informed multi-agent reinforcement learning for supply chains demonstrates how operational constraints and tail-risk objectives can be integrated into coordinated decision policies *[31]*.Censored demand and online adaptation. Censoring biases newsvendor decisions [16], and nonparametric policies that observe only sales can still converge to the optimum [23], [24]. Adaptive conformal inference keeps long-run coverage under arbitrary shift by feeding miscoverage back into the quantile level [25]; ORC carries this argument over to stock-out feedback that arrives only after the lead time.

## III. Problem Formulation

### A. Periodic-Review Lost-Sales System

Series $i$ is an item at a location and $t$ indexes review periods. At the start of period $t$ the orders due arrive ($A_{i,t}$), and the inventory position $\mathrm{IP}_{i,t}$ adds all outstanding orders to the stock on hand. An order raises the position to the level $S_{i,t}$ and arrives after a lead time $\ell_{i,t} \in \{0, \dots, \bar{L}\}$, deterministic or drawn i.i.d.; with $\ell_{i,t} = 0$ it is available at once. Demand $D_{i,t}$ is served from stock and the excess is lost:

$$I_{i,t}^{+} = I_{i,t} + A_{i,t}, \quad o_{i,t} = (S_{i,t} - \mathrm{IP}_{i,t})^{+}, \quad A_{i,t+\ell_{i,t}} \leftarrow A_{i,t+\ell_{i,t}} + o_{i,t} \qquad (1)$$

$$u_{i,t} = (D_{i,t} - I_{i,t}^{+})^{+}, \qquad I_{i,t+1} = (I_{i,t}^{+} - D_{i,t})^{+}. \qquad (2)$$

### B. Cost and Objective

With unit holding cost $h_i$, lost-sale penalty $b_{i,t}$ and promotion indicator $\pi_{i,t}$, the period cost is

$$c_{i,t} = h_i I_{i,t+1} + b_{i,t} u_{i,t}, \qquad b_{i,t} = \frac{h_i \,\mathrm{cf}}{1 - \mathrm{cf}} \left(1 + \kappa \pi_{i,t}\right), \qquad (3)$$

where cf is the base critical fractile and $\kappa$ prices the extra goodwill and advertising loss of a promotional stock-out (Section IV-C discusses where this parameterisation comes from). A policy maps the information available at the start of period $t$ to $S_{i,t}$, and the planner minimises

$$\min \; (1 - w_{\mathrm{out}})\, \mathbb{E}\big[c_{i,t}\big] + w_{\mathrm{out}}\, \mathrm{CVaR}_{\alpha}\big(c_{i,t}\big), \qquad (4)$$

with $\mathrm{CVaR}_{\alpha}(X) = \min_{\eta}\{\eta + \mathbb{E}[(X - \eta)^{+}]/(1 - \alpha)\}$ [17] taken over operating periods pooled across series, in monetary units. The weight $w_{\mathrm{out}}$ is a preference fixed by the planner.

IV. Method

Fig. 1 summarises the framework. A forecaster produces demand quantiles and a risk weight, a differentiable layer turns them into order-up-to levels, and the simulated multi-period cost of these levels is the training signal; ORC adjusts the levels online.

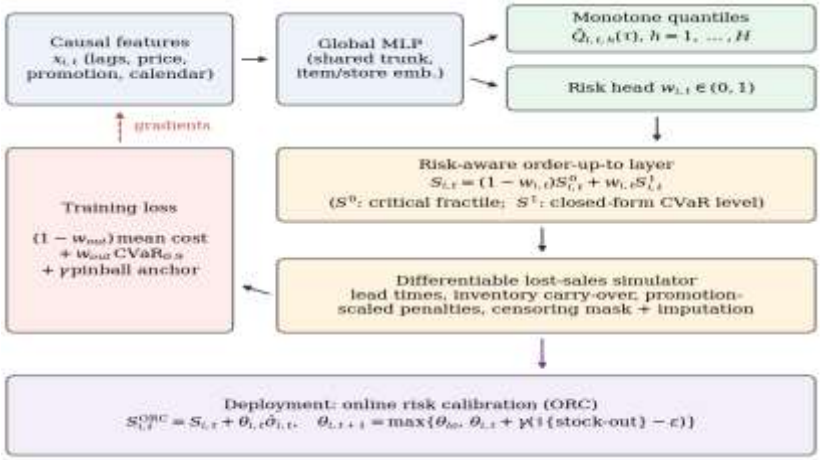



*Figure 1DynRA-DFL. Solid arrows: forward computation; dashed arrow: gradients of the simulated mean–CVaR cost; bottom: online risk calibration at deployment.*

**A. Probabilistic Forecaster**

A global multilayer perceptron with item and store embeddings (two hidden layers of 128 units) receives causal features: eight sales lags, rolling means and standard deviations, recent stock-out frequency, and covariates of the next $H = 3$ periods that are fixed in advance (price relative to its trailing maximum, promotion flags, SNAP days and the event calendar). It outputs quantiles of cumulative demand over $h = 1, \ldots, H$ periods on a grid of $M = 22$ levels $\tau_j \in [0.005, 0.998]$,

$$\hat{Q}_{i,t,h}(\tau_j) = s_{i,t} \sum_{k \le h} \sum_{l \le j} \mathrm{softplus}\left(a_{k,l}(x_{i,t})\right), \qquad (5)$$

where $s_{i,t}$ is the trailing 26-period mean of sales. The double cumulative sum makes the quantiles monotone in both the level and the horizon, so quantile crossing cannot occur. A second head maps the same hidden representation to a risk weight $w_{i,t} \in (0,1)$.

B. Risk-Aware Order-Up-To Layer

Let $X$ denote demand over the protection interval of an order placed in period $t$ (periods $t, \ldots, t + \ell$, a mixture over $\ell$ when lead times are random) with distribution $F$, and consider the protection-interval cost $C(S, X) = h(S - X)^+ + b(X - S)^+$ with $\mathrm{cf} = b/(b + h)$.

**Proposition 1.** *Let $F$ be continuous. (i) The level that minimises* $\mathrm{CVaR}_\alpha[C(S,X)]$ *is*

$$S^1 = (1 - \mathrm{cf})\, F^{-1}\big(\mathrm{cf}(1 - \alpha)\big) + \mathrm{cf}\, F^{-1}\big(1 - (1 - \mathrm{cf})(1 - \alpha)\big). \qquad (6)$$

*(ii) For every $\lambda \in [0,1]$, the objective $(1 - \lambda)\mathbb{E}[C] + \lambda\, \mathrm{CVaR}_\alpha[C]$ has a minimiser between $S^0 = F^{-1}(\mathrm{cf})$ and $S^1$.*

*Proof sketch.* (i) In the Rockafellar–Uryasev representation, the cost exceeds $\eta$ exactly when $X < a = S - \eta/h$ or $X > c = S + \eta/b$. Stationarity in $\eta$ makes the two tails carry probability $1 - \alpha$ together, and stationarity in $S$ balances $h$ times the lower-tail mass against $b$ times the upper-tail mass, which gives masses $\mathrm{cf}(1 - \alpha)$ and $(1 - \mathrm{cf})(1 - \alpha)$; equal cost at both boundaries, $h(S - a) = b(c - S)$, then yields (6). (ii) Both terms are convex in $S$; below $\min(S^0, S^1)$ both are non-increasing and above $\max(S^0, S^1)$ both are non-decreasing. ▫

The layer evaluates $S^0$ and (6) on the predicted quantiles by linear interpolation and sets

$$S_{i,t} = \big(1 - w_{i,t}\big)\, \hat{S}^0_{i,t} + w_{i,t}\, \hat{S}^1_{i,t}. \qquad (7)$$

Because $w_{i,t}$ is learned and can reach every point of the segment, part (ii) implies that the mixture loses nothing relative to solving the mean–CVaR problem for a fixed weight. In promotional periods the fractile uses the penalty of the period in which the order arrives. As a

numerical check, for $X \sim \mathcal{N}(10, 2^2)$, $\mathrm{cf} = 0.85$ and $\alpha = 0.9$, (6) gives 13.27, within 0.01 of a grid search on $4 \times 10^5$ samples.

**C. Differentiable Simulation and Censoring Correction**

Recursion (1)–(3) involves only sums, minima and positive parts, so the cost of a window of periods is differentiable almost everywhere in the order-up-to levels and, through (5)–(7), in the network weights; gradients are taken pathwise on historical demand traces, as in hindsight differentiable policy optimisation [13]. Our cost model and censoring treatment build upon the single-period risk-aware layer of Zhang et al. [8]: we adopt its promotion-scaled lost-sale penalty with $\kappa = 0.5$, its promotion and stock-out proxies, and its principle of correcting the decision loss rather than the forecast, and we extend its masked overage loss from one period to carry-over dynamics. Observed sales $y_{i,t}$ understate demand in flagged periods ($\xi_{i,t} = 1$), which inflates simulated leftovers twice: in the holding cost of that period and in the stock carried into the next one. We therefore train on

$$\tilde{c}_{i,t} = h_i\big(1 - \xi_{i,t}\big) I_{i,t+1} + b_{i,t} u_{i,t}, \qquad \tilde{I}_{i,t+1} = \big(I_{i,t}^{+} - \max\{y_{i,t}, \hat{m}_{i,t}\}\big)^{+} \text{ if } \xi_{i,t} = 1, \tag{8}$$

where $\hat{m}_{i,t}$ is the predicted one-period median (gradient stopped). Masking removes the overage term that a truncated observation overstates, as in the single-period case, and the imputation keeps phantom stock from suppressing simulated stock-outs in later periods.

**D. Training Objective and Procedure**

Training has two phases. Phase 1 fits (5) with the pinball loss over all horizons [26], switched to its one-sided (Tobit-style) form on censored targets. Phase 2 unrolls the simulator over windows of four warm-up and sixteen loss periods for batches of 128 series and minimises

$$\mathcal{L} = \frac{(1 - w_{\mathrm{out}})\,\overline{\tilde{c}} + w_{\mathrm{out}}\,\mathrm{CVaR}_{0.9}(\tilde{c})}{\mathrm{sg}\big(\overline{\tilde{c}}\big)} + \gamma\,\mathcal{L}_{\mathrm{pin}}, \tag{9}$$

where the batch-mean normaliser $\mathrm{sg}(\cdot)$ is excluded from differentiation, so that the risk functional acts on monetary costs, and the pinball anchor ($\gamma = 0.3$) keeps the quantiles interpretable. Initial stock equals the first order-up-to level and is also excluded from differentiation. We use Adam [27] with learning rate $3 \times 10^{-4}$, 50 steps per epoch and at most 25 epochs, and stop early on the validation objective of the policy simulated with observed sales. Everything is implemented in JAX [28].

**E. Online Risk Calibration**

Offline training fixes the map from features to levels, and under drift that map goes stale. ORC adds a per-series margin in units of the predicted spread $\hat{\sigma}_{i,t} = \hat{Q}(0.9) - \hat{Q}(0.5)$ of protection-interval demand:

$$S_{i,t}^{\mathrm{ORC}} = (S_{i,t} + \theta_{i,t}\hat{\sigma}_{i,t})^{+}, \quad \theta_{i,t+1} = \max\{\theta_{lo},\, \theta_{i,t} + \gamma\big(\mathbb{1}\{u_{i,t} > 0\} - \varepsilon\big)\}. \tag{10}$$

The update needs only whether a stock-out occurred, which is observable even when demand is censored. With $\varepsilon = 1 - \mathrm{cf}$ it is an online subgradient step on the newsvendor cost of the margin, the mechanism behind nonparametric censored-demand policies [23]; in our experiments $\varepsilon$ is the stock-out frequency of the offline policy on the validation window, so that ORC is neutral when nothing drifts.

**Proposition 2.** *Suppose* $0 \le D_{i,t} \le \bar{D}$, $\hat{\sigma}_{i,t} \ge \sigma_{\min} > 0$, *every order arrives within* $\bar{L}$ *periods, and let* $\theta^{\star} = (\bar{L} + 1)\bar{D}/\sigma_{\min}$. *Then for any demand sequence and any forecaster,* $\theta_{i,t} \le B := \max\{\theta_{i,1}, \theta^{\star}\} + (\bar{L} + 1)\gamma$ *for all* $t$, *and*

$$\frac{1}{T}\sum_{t=1}^{T} \mathbb{1}\{u_{i,t} > 0\} \le \varepsilon + \frac{B - \theta_{i,1}}{\gamma T}. \tag{11}$$

*Proof sketch.* The position after ordering in period $t$ is at least $S_{i,t}^{\mathrm{ORC}}$, and all of it is on hand by $t+\bar{L}$ except what has been sold, so a stock-out in period $t'$ implies that demand over periods $t'-\bar{L},\dots,t'$ exceeded $S_{i,t'-\bar{L}}^{\mathrm{ORC}}$, which is impossible if $\theta_{i,t'-\bar{L}} \geq \theta^{\star}$. Hence every increase of $\theta$ at $t'$ starts from a value below $\theta^{\star}$ at most $\bar{L}+1$ steps earlier, each step adding at most $\gamma$, which bounds $\theta$ by $B$. The lower clip only adds upward mass, so telescoping (10) gives $\gamma\sum_t \left(\mathbb{1}\{u_{i,t}>0\}-\varepsilon\right) \leq \theta_{i,T+1}-\theta_{i,1} \leq B-\theta_{i,1}$. ▫

The guarantee is one-sided by design: intermittent zero demand can prevent stock-outs however low the margin goes, so only the upper bound on stock-out frequency, which is what a service target requires, holds without assumptions on demand. It is the stock-out analogue of adaptive conformal inference [25], extended to feedback delayed by the lead time.

**F. Complexity**

A training step costs $O\left(BW(HM+d^2)\right)$ for $B$ series, $W$ periods and hidden width $d$, and the simulator adds $O(BW\bar{L})$. On a single CPU core, phase 1 takes about 25 s on the M5 panel, a phase-2 run 20–40 s, and evaluating all 909 series over the test horizon well under a second.

## V. Experimental Setup

### A. Data

Table I summarises the two public panels. *M5* [1] contains daily unit sales, weekly posted prices and an event and SNAP calendar for ten Walmart stores. We aggregate to Walmart weeks, take the best-selling items of five departments, and keep item–store series that are listed throughout and whose longest run of zero-sales weeks is at most four (longer runs indicate delisting rather than stock-outs). A week is promotional if its price is at least 5% below the trailing 26-week maximum, and censored if it has at least three zero-sales days (Christmas excluded) or no sales while the item is listed, which are the proxies adopted in Section IV-C; we additionally require the number of zero days to be implausible under the trailing daily sales rate (binomial tail $p<0.01$) so that slow movers do not raise spurious flags. *Stallion* is a monthly beverage panel distributed with the pytorch-forecasting library; we keep the agency–SKU series with positive volume in at least 90% of the training months, treat discounts of at least 5% as promotions, and flag zero months that follow positive trailing volume.

*Table 1Data sets and protocol*

| | **M5 (Walmart)** | **Stallion** |
|---|---|---|
| **Period / series** | week / 909 item–store | month / 281 agency–SKU |
| **Items, locations** | 186 items, 10 stores | 12 SKUs, 57 agencies |
| **Train / val. / test** | 30–199 / 200–219 / 220–276 | 12–41 / 42–47 / 48–59 |
| **Promotion rate** | 4.2% | 85.9% |
| **Stock-out proxy rate** | 8.2% | 0.47% |
| **Lead time (main)** | 1 week | 1 month |

### B. Protocol

Splits are chronological and all features are causal; posted prices, promotion flags and calendars of the next $H$ periods are treated as known. Every policy is simulated from the start of the validation window, so the state at the first test period is generated by the policy itself, and costs are counted over the 57 test weeks of M5 and the 12 test months of Stallion. Unit cost is 0.70 of the median posted price, holding cost is 2% of unit cost per period, $\mathrm{cf}=0.85$ and $\kappa=0.5$, as in the setting of Section IV-C; the main lead time is one period and Section VI-

F varies it. Because true demand is unobserved in censored periods, we report two protocols. The *observed* protocol uses sales as demand, the standard choice, which systematically favours policies that under-order where stock-outs are frequent. The *reconstructed* protocol (superscript $R$) fills each zero-sales open day of a flagged M5 week with the trailing 8-week daily rate, which raises test-period demand by 3.3%; it is identical for all methods and used only for evaluation.

**C. Methods**

All methods share the simulator, the cost parameters and the initial state and differ only in how $S_{i,t}$ is produced. The baselines are a moving-average base-stock with normal safety stock; LightGBM [29] with an L2 objective plus empirical residual quantiles (LGB-L2); LightGBM quantile regression at eleven levels interpolated onto the quantile grid (LGB-QR); the network of Section IV-A trained with the pinball loss (MLP-QR) or its censoring-aware form (MLP-CQR) and used as PTO forecasters at the critical fractile; the same PTO forecasters with a CVaR decision ($w = 0.5$ in (7)); *static RA-DFL*, which trains the same network on the mean–CVaR newsvendor cost of protection-interval demand with masked overage and a learned risk weight, i.e. the single-period layer of Section IV-C adapted to one location; and *Dyn-DFL*, our dynamic training with $w_{\text{out}} = 0$ and $w \equiv 0$. *DynRA-DFL* uses $w_{\text{out}} = 0.3$ and the learned risk weight; *DynRA-DFL+ORC* adds (10) with $\theta_{lo} = -1.5$ and $\gamma$ selected on validation from $\{0.02, 0.05, 0.1, 0.2\}$. As an oracle-tuned reference, *Hindsight const.* gives each series the constant level that minimises its own test cost.

**D. Metrics and Tests**

Cost is the mean cost per series-period. $\text{CVaR}_s$ averages over series the $\text{CVaR}_{0.9}$ of each series' period costs (the bad weeks of an item), $\text{CVaR}_w$ is the $\text{CVaR}_{0.8}$ of total cost per test period (the bad weeks of the portfolio), and the fill rate is the share of demand served. Neural methods are trained with three seeds and reported as means. Significance is assessed with an item-clustered paired bootstrap of the mean cost difference (4,000 resamples, seed-averaged costs) and with a Diebold–Mariano test on period totals with a Newey–West variance [30].

**VI. Results**

**A. Main Comparison**

*Table 2M5 test results (mean of three seeds; superscript R: reconstructed demand; *oracle-tuned on test demand)*

| Method | Cost | $\text{CVaR}_s$ | $\text{CVaR}_w$ | Fill % | $\text{Cost}^R$ | $\text{CVaR}_s^R$ |
|---|---|---|---|---|---|---|
| **MA + normal SS** | 1.103 | 3.58 | 1216 | 94.2 | 1.014 | 3.53 |
| **LGB-L2 + emp. q.** | 1.128 | 3.17 | 1176 | 96.1 | 1.047 | 3.12 |
| **LGB-QR** | 0.984 | 3.04 | 1016 | 95.9 | 0.896 | 2.96 |
| **LGB-QR + CVaR** | 1.132 | 2.82 | 1164 | 97.8 | 1.022 | 2.58 |
| **MLP-QR** | 1.042 | 3.16 | 1090 | 95.8 | 0.955 | 3.10 |
| **MLP-QR + CVaR** | 1.186 | 2.98 | 1242 | 97.8 | 1.070 | 2.76 |
| **MLP-CQR** | 1.080 | 3.21 | 1152 | 96.3 | 0.967 | 2.96 |
| **Static RA-DFL** | 1.144 | 3.10 | 1218 | 97.2 | 1.030 | 2.83 |
| **Dyn-DFL** | 1.017 | 3.39 | 1092 | 94.6 | 0.914 | 3.26 |
| **DynRA-DFL** | 1.050 | 3.13 | 1107 | 96.1 | 0.940 | 2.93 |
| **DynRA-DFL+ORC** | 1.051 | 3.11 | 1108 | 96.2 | 0.942 | 2.89 |
| **Hindsight const.*** | 1.050 | 3.37 | 1139 | 94.6 | 0.985 | 3.30 |

The risk-neutral dynamic variant, Dyn-DFL, is the cheapest neural policy (Table II). Its mean cost is 11.2% lower than that of static RA-DFL and 5.9% lower than that of its own PTO initialisation (MLP-CQR) on observed sales, and 11.3% lower and 5.6% lower, respectively, on reconstructed demand. The single-period loss shows why carry-over dynamics must enter training: static RA-DFL prices every leftover as if it were charged once, overstocks (fill rate 97.17%) and costs more than every risk-neutral PTO pipeline, including the moving-average heuristic. DynRA-DFL trades part of the mean-cost gain for a lower tail: relative to Dyn-DFL its $\mathrm{CVaR}_s$ is 7.9% lower on observed sales and 10.4% lower on reconstructed demand, at a mean-cost premium of 3.3% and 2.9%. The reduction is specific to the item-level tail: the portfolio tail $\mathrm{CVaR}_w$, which follows total cost, is 1.4% higher on observed sales and 0.2% higher on reconstructed demand. Against MLP-CQR, the forecaster it starts from, DynRA-DFL is 2.8% lower in mean cost on observed sales and 2.9% lower on reconstructed demand, with a $\mathrm{CVaR}_s^R$ 1.2% lower. The cheapest policy on average is LightGBM-QR, whose mean cost is 3.2% lower than that of Dyn-DFL on observed sales and 1.9% lower on reconstructed demand. The advantage comes from the forecaster class: its test WAPE is 0.216, against 0.227 for MLP-QR and 0.236 for DynRA-DFL, and DFL changes where a network spends its accuracy, not how much accuracy the network class can reach. DynRA-DFL nevertheless attains a $\mathrm{CVaR}_s^R$ 1.3% lower than LightGBM-QR, at a mean cost 4.9% higher. LightGBM-QR retains the lower $\mathrm{CVaR}_s$ on observed sales and the lower portfolio tail $\mathrm{CVaR}_w$ under both protocols, so the comparison is a trade-off rather than dominance. DynRA-DFL therefore trades average inventory cost against targeted tail-risk reductions. Adaptivity itself matters: even the constant level tuned per series on the test demand has a mean cost 3.3% higher than Dyn-DFL. Seed variation is small (standard deviation of cost 0.001 for DynRA-DFL and 0.003 for Dyn-DFL).

**B. Statistical Significance**

*Table 3Paired mean-cost difference (baseline − DynRA-DFL) on M5; item-clustered bootstrap on seed-averaged costs*

| **Baseline** | **$\Delta$ obs. [95% CI]** | ***p*** | **$\Delta$ rec. [95% CI]** | ***p*** |
|---|---|---|---|---|
| **LGB-QR** | −0.066 [−0.088, −0.049] | <0.001 | −0.044 [−0.056, −0.034] | <0.001 |
| **LGB-QR + CVaR** | +0.082 [+0.056, +0.112] | <0.001 | +0.083 [+0.058, +0.110] | <0.001 |
| **MLP-QR** | −0.008 [−0.019, +0.001] | 0.093 | +0.016 [+0.005, +0.031] | <0.001 |
| **MLP-QR + CVaR** | +0.136 [+0.113, +0.162] | <0.001 | +0.130 [+0.107, +0.156] | <0.001 |
| **MLP-CQR** | +0.030 [+0.019, +0.045] | <0.001 | +0.028 [+0.019, +0.038] | <0.001 |
| **Static RA-DFL** | +0.094 [+0.080, +0.109] | <0.001 | +0.090 [+0.077, +0.105] | <0.001 |
| **Dyn-DFL** | −0.034 [−0.041, −0.026] | <0.001 | −0.026 [−0.033, −0.019] | <0.001 |

Table III makes the comparisons precise. On observed sales, DynRA-DFL is significantly cheaper than LightGBM-QR+CVaR, MLP-QR+CVaR, MLP-CQR and static RA-DFL, it is significantly more expensive than LightGBM-QR and Dyn-DFL and it is statistically indistinguishable from MLP-QR. On reconstructed demand, DynRA-DFL is significantly cheaper than LightGBM-QR+CVaR, MLP-QR, MLP-QR+CVaR, MLP-CQR and static RA-

DFL and it is significantly more expensive than LightGBM-QR and Dyn-DFL. The Diebold–Mariano tests on weekly totals reach the same conclusion at the 5% level in 21 of the 22 comparisons. The pattern is the one Fig. 2 makes explicit: DynRA-DFL buys item-level tail protection with mean cost relative to the risk-neutral policies, and the protocol decides how a correction for censoring is credited.

**C. Mean–Risk Frontier and the CVaR-on-PTO Reversal**

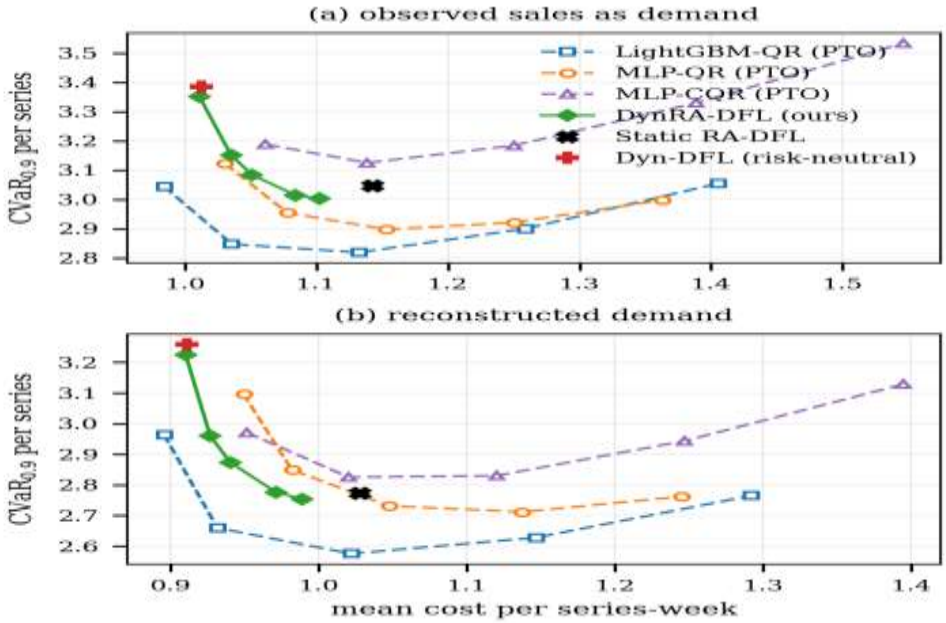



*Figure 2Mean–risk frontiers on M5 (seed 0). PTO frontiers vary the decision-level weight w∈{0,0.25,0.5,0.75,1}; DynRA-DFL varies the training-level preference w_out∈{0,0.15,0.3,0.5,0.7}. Down and to the left is better.*

Fig. 2 compares whole frontiers. Adding a CVaR decision to PTO forecasts raises mean cost monotonically in the decision weight, at $w = 0.5$ by 13.9% for MLP-QR and 15.0% for LightGBM-QR, and the portfolio tail $\mathrm{CVaR}_w$ rises as well (13.9% and 14.5%). This reproduces, in a dynamic system with carry-over, the reversal that Zhang et al. [8] documented for single-period multi-channel allocation, and it confirms their diagnosis that risk aversion has to be learned jointly with the forecast. The item-level tail $\mathrm{CVaR}_s$ does fall (5.7% lower for MLP-QR at $w = 0.5$), so CVaR decisions buy item-level protection with a large increase in stock. Whether training-level risk aversion is the more efficient lever depends on the protocol. Under reconstructed demand, four of the five DynRA-DFL points lie below the MLP-QR frontier, four below the MLP-CQR frontier and none below the LightGBM-QR frontier. Under observed sales, none of the five DynRA-DFL points lie below the MLP-QR frontier, four below the MLP-CQR frontier and none below the LightGBM-QR frontier. The LightGBM-QR frontier reflects its stronger forecasts rather than a better decision layer.

**D. Ablations**

*Table 4Ablations on M5 (mean of three seeds); p^R: bootstrap p-value of the cost difference to the full model under reconstructed demand*

| Variant | Cost | $\mathrm{CVaR}_s$ | $\mathrm{Cost}^R$ | $\mathrm{CVaR}_s^R$ | $\mathrm{CVaR}_w^R$ | $p^R$ |
|---|---|---|---|---|---|---|
| **DynRA-DFL (full)** | 1.050 | 3.13 | 0.940 | 2.93 | 966 | – |
| **global risk weight** | 1.055 | 3.09 | 0.945 | 2.87 | 971 | <0.001 |
| **no risk layer ($w = 0$)** | 1.053 | 3.09 | 0.944 | 2.87 | 968 | 0.002 |
| **no training CVaR** | 1.015 | 3.37 | 0.913 | 3.24 | 962 | <0.001 |

| | | | | | | |
|---|---|---|---|---|---|---|
| **no censoring corr.** | 1.028 | 3.18 | 0.954 | 3.16 | 1009 | 0.132 |
| **mask, no imputation** | 1.043 | 3.13 | 0.936 | 2.94 | 963 | <0.001 |
| **no pinball anchor** | 1.051 | 3.13 | 0.940 | 2.92 | 967 | 0.230 |
| **static training** | 1.144 | 3.10 | 1.030 | 2.83 | 1065 | <0.001 |

Table IV isolates the components. Removing the training-level CVaR recovers the risk-neutral trade-off: $\mathrm{CVaR}_s^R$ is 10.7% higher and mean cost 2.9% lower. The censoring correction is where the two protocols disagree: without it, mean cost is 2.1% lower on observed sales but 1.6% higher on reconstructed demand ($p^R$ = 0.132), and $\mathrm{CVaR}_s^R$ is 7.9% higher; correcting for censoring raises stock exactly where the observed protocol cannot see the lost demand. Masking without imputing the carried-over state gives a mean cost 0.4% lower and a $\mathrm{CVaR}_s^R$ 0.7% higher under reconstructed demand, so the mask carries most of the effect. Neither the state-conditional weight nor the CVaR mixture in the layer is needed for the tail reduction: with a single global weight $\mathrm{CVaR}_s^R$ is 1.8% lower, and without the mixture 1.8% lower, so the reduction comes from the training-level preference; removing the pinball anchor leaves $\mathrm{CVaR}_s^R$ 0.1% lower but discards the probabilistic meaning of the quantiles. Static training is the costliest variant under both protocols.

**E. What the Risk Head Learns**

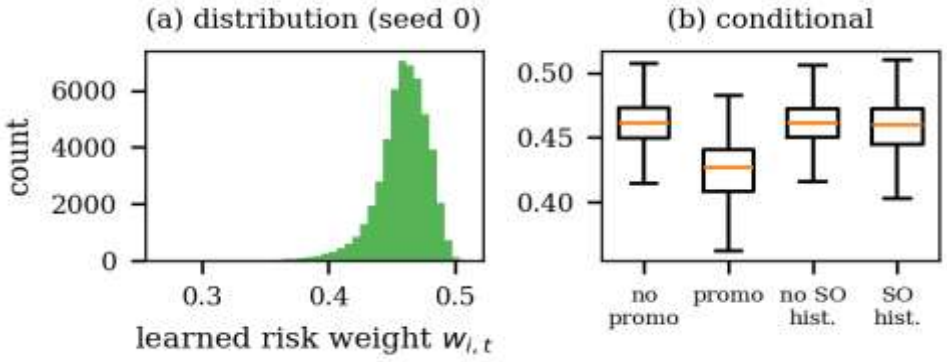



*Figure 3Learned risk weight of DynRA-DFL on the M5 test period: (a) distribution; (b) by promotion in the protection interval and by stock-out history.*

The learned weight has mean 0.458 and standard deviation 0.022 (Fig. 3), close to the value that the global-weight ablation converges to, which explains why replacing it costs little. Its state dependence is statistically clear but economically small: it is lower in promotional protection intervals (0.414 versus 0.459, $t = -77.6$) and slightly lower after recent stock-outs (0.455 versus 0.459, $t = -27.8$). The promotional effect runs against the intuition that risk aversion is worth more in promotion weeks; in our layer the promotional penalty already raises the critical fractile, and the learned weight partly offsets that increase rather than compounding it.

**F. Lead Time and Cost Ratio**

*Table 5Sensitivity on M5 (seed 0, observed sales): mean cost and CVaR_s of Dyn-DFL / DynRA-DFL*

| Setting | MLP-CQR | Static | Dyn-DFL | DynRA | +ORC | $\mathrm{CVaR}_s$ Dyn / RA |
|---|---|---|---|---|---|---|
| $L = 0$ | 0.650 | 0.669 | 0.641 | 0.664 | 0.664 | 2.06 / 1.88 |
| $L = 1$ (main) | 1.062 | 1.143 | 1.013 | 1.051 | 1.052 | 3.39 / 3.08 |
| $L = 2$ | 1.499 | 1.670 | 1.310 | 1.362 | 1.364 | 4.36 / 4.06 |
| $L \in \{1, 2\}$ | 1.769 | 1.849 | 1.670 | 1.727 | 1.730 | 4.84 / 4.50 |

| | | | | | | |
|---|---|---|---|---|---|---|
| **cf = 0.70** | 0.748 | 0.792 | 0.695 | 0.709 | 0.709 | 2.07 / 2.04 |
| **cf = 0.95** | 1.602 | 1.690 | 1.546 | 1.621 | 1.621 | 5.87 / 4.76 |

Table V varies the lead time, including random lead times, and the cost ratio; every row retrains all methods. Dyn-DFL is the cheapest policy in all six settings, dynamic training beats static training in all six and the PTO forecaster in all six, and the training-level CVaR lowers $\mathrm{CVaR}_s$ relative to Dyn-DFL in all six. Costs grow with the deterministic lead time for every method because the protection interval lengthens. The cost ratio shifts absolute levels as it must, but the ranking of training principles is stable. Think4SCND offers a related dynamic supply-chain example of reinforcement learning coupled to sequential planning and network-level decisions [32].

**G. Demand Drift and Online Risk Calibration**

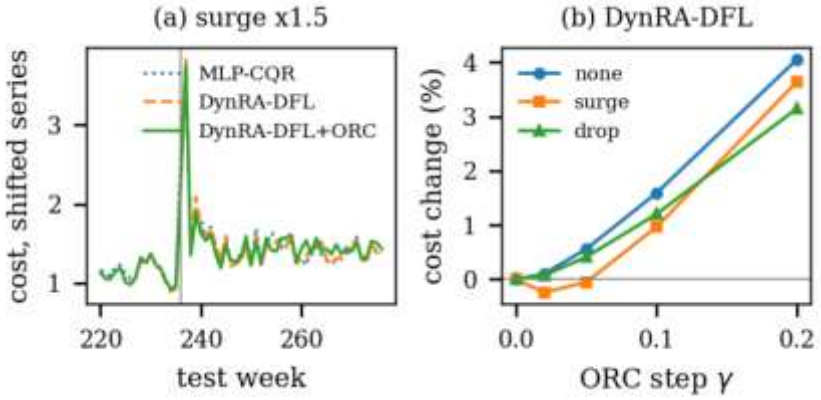



*Figure 4Stress test: half of the M5 series receive a persistent level shift from week 236; models are not retrained. (a) Mean weekly cost of the shifted series under a ×1.5 surge; (b) change in total test cost of DynRA-DFL as a function of the ORC step γ.*

*Table 6Stress test on M5 (mean over seeds): post-shift cost of the shifted series without ORC, change with ORC (γ=0.1), fill rate of the shifted series, and change in total cost without a shift*

| Policy | Surge cost | Surge Δ% | Fill: off → ORC | Drop Δ% | No shift Δ% |
|---|---|---|---|---|---|
| **MLP-CQR** | 1.533 | +1.3 | 93.7 → 95.2 | −1.1 | +1.5 |
| **LGB-QR** | 1.454 | +1.4 | 93.3 → 94.8 | −0.5 | +2.1 |
| **Dyn-DFL** | 1.544 | −2.5 | 91.7 → 93.6 | +0.4 | +1.1 |
| **DynRA-DFL** | 1.523 | +0.1 | 93.3 → 95.0 | +0.1 | +1.6 |

Without drift ORC is nearly neutral: validation selects the smallest step, and on the M5 test period it changes DynRA-DFL's mean cost by 0.1% higher while $\mathrm{CVaR}_s^R$ becomes 1.2% lower. To probe drift, a random half of the series receive a persistent ×1.5 surge or ×0.6 drop from test week 236; features are rebuilt causally from the shifted sales and no model is retrained. Under the surge, ORC with $\gamma = 0.1$ raises the post-shift fill rate of the shifted series for every policy (Table VI), but its cost effect is small and mixed: the post-shift cost of the shifted series changes by +0.1% for DynRA-DFL, −2.5% for Dyn-DFL, +1.3% for MLP-CQR and +1.4% for LightGBM-QR. With a penalty-to-holding ratio of 5.7, the stock added after stock-outs

costs roughly as much as the lost sales it prevents; the best step on the grid, $\gamma = 0.05$, changes DynRA-DFL's post-shift cost by −0.9%. Under the drop the changes are +0.1%, +0.4%, −1.1% and −0.5%, since without stock-outs the margin falls by only $\gamma\varepsilon$ per period. In this cost setting ORC is therefore a service safeguard backed by Proposition 2 rather than a cost-reduction device.

**H. Second Data Set: Stallion**

*Table 7Stallion test results (12 months; cost and CVaR_s in thousands, CVaR_w in millions)*

| Method | Cost (k) | $\mathbf{CVaR}_s$ (k) | $\mathbf{CVaR}_w$ (M) | Fill % | seeds |
|---|---|---|---|---|---|
| **MA + normal SS** | 24.90 | 68.9 | 9.37 | 96.82 | 1 |
| **LGB-QR** | 22.71 | 52.1 | 7.56 | 98.37 | 1 |
| **LGB-QR + CVaR** | 44.74 | 65.4 | 14.32 | 99.76 | 1 |
| **MLP-QR** | 28.58 | 67.3 | 10.48 | 97.85 | 3 |
| **MLP-QR + CVaR** | 40.07 | 70.8 | 13.09 | 99.05 | 3 |
| **MLP-CQR** | 28.13 | 64.1 | 9.96 | 98.15 | 3 |
| **Static RA-DFL** | 27.52 | 66.2 | 10.48 | 97.78 | 3 |
| **Dyn-DFL** | 26.44 | 71.0 | 10.69 | 97.07 | 3 |
| **DynRA-DFL** | 26.17 | 65.7 | 10.16 | 97.60 | 3 |
| **DynRA-DFL+ORC** | 26.16 | 65.4 | 10.11 | 97.65 | 3 |
| **Hindsight const.*** | 17.00 | 40.6 | 6.29 | 98.76 | 1 |

Stallion has frequent promotions and almost no censoring, and only twelve test months. DynRA-DFL's mean cost is 8.4% lower than MLP-QR, 7.0% lower than MLP-CQR, 1.0% lower than Dyn-DFL, 4.9% lower than static RA-DFL and 15.2% higher than LightGBM-QR (Table VII). The bootstrap separates DynRA-DFL from MLP-QR, MLP-CQR, LightGBM-QR, static RA-DFL, MLP-QR+CVaR and LightGBM-QR+CVaR, but not from Dyn-DFL and the variant without censoring correction. Adding a CVaR decision to PTO forecasts again raises cost (40.2% for MLP-QR and 97.0% for LightGBM-QR). With so few periods and little censoring, Stallion mainly tests the promotion mechanism, and the censoring correction is, as expected, immaterial there.

**VII. Discussion**

Three findings carry beyond this architecture. First, the dynamics belong in the training loss: a single-period decision loss, even a risk-aware one, misprices leftovers in a system that carries them, and on M5 it is worse than plain PTO. Second, risk aversion should be expressed in training rather than bolted onto the decision, which extends the single-period evidence to lead times and carry-over; the training-level preference gives a controllable mean–tail frontier, whereas the state-conditional weight contributes little. Moving along the frontier lowers item-level tail cost at a higher mean cost, so the risk preference becomes an explicit planning lever. Third, how censored periods are evaluated changes conclusions: the observed-sales protocol penalises corrections for censoring by construction, and the reconstructed protocol, while itself a proxy, shows those corrections lowering tail cost.

Several extensions follow. Gradient-boosted trees remain the strongest forecaster class on M5, so coupling tree ensembles with decision-focused training through out-of-fold base predictions is a natural next step; promotion and stock-out proxies can be replaced by recorded flags where available; and the single-echelon system can be extended to multi-echelon networks with supplier lead-time uncertainty.

## VIII. Conclusion

DynRA-DFL trains a probabilistic demand forecaster through a differentiable lost-sales inventory simulator with lead times, a closed-form risk-aware order-up-to layer and a censoring-corrected mean–CVaR cost, and adapts online through a stock-out-driven calibration rule with a distribution-free service guarantee. On two real retail panels, the risk-neutral dynamic variant yields the cheapest neural policies, dynamic training clearly outperforms single-period decision losses, and risk added only at the decision stage raises cost, whereas online calibration raises service under demand surges at small cost. DynRA-DFL trades average inventory cost against targeted tail-risk reductions, and the training-level preference $w_{\text{out}}$ is the planner's lever for choosing a point on this trade-off. Combining tree-ensemble forecasters with dynamic decision-focused training is the most direct next step for lowering mean cost further.

## References


D. Teng, "TRIAD: Deadline- and reliability-aware scheduling for post-earthquake structural assessment workflows," World Journal of Engineering Research, vol. 3, no. 6, pp. 55–66, 2025.

A. N. Elmachtoub and P. Grigas, "Smart 'predict, then optimize'," *Manage. Sci.*, vol. 68, no. 1, pp. 9–26, 2022, doi: 10.1287/mnsc.2020.3922.

P. Donti, B. Amos, and J. Z. Kolter, "Task-based end-to-end model learning in stochastic optimization," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 30, 2017, pp. 5484–5494.

J. Mandi, J. Kotary, S. Berden, M. Mulamba, V. Bucarey, T. Guns, and F. Fioretto, "Decision-focused learning: Foundations, state of the art, benchmark and future opportunities," *J. Artif. Intell. Res.*, vol. 80, pp. 1623–1701, 2024, doi: 10.1613/jair.1.15320.

U. Sadana, A. Chenreddy, E. Delage, A. Forel, E. Frejinger, and T. Vidal, "A survey of contextual optimization methods for decision-making under uncertainty," *Eur. J. Oper. Res.*, vol. 320, no. 2, pp. 271–289, 2025, doi: 10.1016/j.ejor.2024.03.020.

G.-Y. Ban and C. Rudin, "The big data newsvendor: Practical insights from machine learning," *Oper. Res.*, vol. 67, no. 1, pp. 90–108, 2019, doi: 10.1287/opre.2018.1757.

A. Oroojlooyjadid, L. V. Snyder, and M. Takáč, "Applying deep learning to the newsvendor problem," *IISE Trans.*, vol. 52, no. 4, pp. 444–463, 2020, doi: 10.1080/24725854.2019.1632502.

X. Zhang, Y. Yang, and Y. Wang, "Risk-aware decision-focused demand forecasting for multi-channel inventory optimization under promotion and stock-out constraints," *Highlights Bus. Econ. Manag.*, vol. 69 (MSIED 2026), pp. 263–279, 2026.

M. Qi, Y. Shi, Y. Qi, C. Ma, R. Yuan, D. Wu, and Z.-J. Shen, "A practical end-to-end inventory management model with deep learning," *Manage. Sci.*, vol. 69, no. 2, pp. 759–773, 2023, doi: 10.1287/mnsc.2022.4564.

J. Gijsbrechts, R. N. Boute, J. A. Van Mieghem, and D. J. Zhang, "Can deep reinforcement learning improve inventory management? Performance on lost sales, dual-sourcing, and multi-echelon problems," *Manuf. Serv. Oper. Manag.*, vol. 24, no. 3, pp. 1349–1368, 2022, doi: 10.1287/msom.2021.1064.

J. Chen, X. Han, X. Zhang, and Y. Yang, “Exact concept-level explanations for early detection of financial misstatements,” Eurasia Journal of Science and Technology, vol. 5, no. 2, pp. 34–48, 2023.

D. Madeka, K. Torkkola, C. Eisenach, A. Luo, D. P. Foster, and S. M. Kakade, “Deep inventory management,” arXiv:2210.03137, 2022.

M. Alvo, D. Russo, and Y. Kanoria, “Neural inventory control in networks via hindsight differentiable policy optimization,” arXiv:2306.11246, 2023, doi: 10.48550/arXiv.2306.11246.

P. Zipkin, “Old and new methods for lost-sales inventory systems,” *Oper. Res.*, vol. 56, no. 5, pp. 1256–1263, 2008, doi: 10.1287/opre.1070.0471.

J. Wang, Y. Ge, J. Liu, and B. Jiang, “Early warning of liquidity stress in U.S. banks with seasonally aligned frailty hazards and delayed-feedback conformal alarms,” Journal of Trends in Finance and Economics, vol. 1, no. 1, pp. 85–100, 2024.

O. Besbes and A. Muharremoglu, “On implications of demand censoring in the newsvendor problem,” *Manage. Sci.*, vol. 59, no. 6, pp. 1407–1424, 2013, doi: 10.1287/mnsc.1120.1654.

R. T. Rockafellar and S. Uryasev, “Optimization of conditional value-at-risk,” *J. Risk*, vol. 2, no. 3, pp. 21–41, 2000, doi: 10.21314/JOR.2000.038.

J. Gotoh and Y. Takano, “Newsvendor solutions via conditional value-at-risk minimization,” *Eur. J. Oper. Res.*, vol. 179, no. 1, pp. 80–96, 2007.

X. Chen, M. Sim, D. Simchi-Levi, and P. Sun, “Risk aversion in inventory management,” *Oper. Res.*, vol. 55, no. 5, pp. 828–842, 2007, doi: 10.1287/opre.1070.0429.

S. Ahmed, U. Çakmak, and A. Shapiro, “Coherent risk measures in inventory problems,” *Eur. J. Oper. Res.*, vol. 182, no. 1, pp. 226–238, 2007.

A. Ruszczyński, “Risk-averse dynamic programming for Markov decision processes,” *Math. Program.*, vol. 125, no. 2, pp. 235–261, 2010, doi: 10.1007/s10107-010-0393-3.

J. Chen, X. Zhang, and X. Han, “Explainable identification of tax-related financial reporting anomalies in U.S. public companies: Cross-fitted tax-consistency residuals and Owen-value Shapley propagation,” World Journal of Economics and Business Research, vol. 1, no. 1, pp. 33–45, 2023.

W. T. Huh and P. Rusmevichientong, “A nonparametric asymptotic analysis of inventory planning with censored demand,” *Math. Oper. Res.*, vol. 34, no. 1, pp. 103–123, 2009, doi: 10.1287/moor.1080.0355.

W. T. Huh, R. Levi, P. Rusmevichientong, and J. B. Orlin, “Adaptive data-driven inventory control with censored demand based on Kaplan–Meier estimator,” *Oper. Res.*, vol. 59, no. 4, pp. 929–941, 2011, doi: 10.1287/opre.1100.0906.

I. Gibbs and E. Candès, “Adaptive conformal inference under distribution shift,” in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 34, 2021, pp. 1660–1672.

R. Koenker and G. Bassett, “Regression quantiles,” *Econometrica*, vol. 46, no. 1, pp. 33–50, 1978, doi: 10.2307/1913643.

D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2015, arXiv:1412.6980.

J. Wang, J. Liu, Y. Ge, and B. Jiang, “Risk-aware ensemble learning for credit default prediction under economic uncertainty,” World Journal of Economics and Business Research, vol. 1, no. 1, pp. 18–32, 2023.

G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, “LightGBM: A highly efficient gradient boosting decision tree,” in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 30, 2017, pp. 3146–3154.

F. X. Diebold and R. S. Mariano, “Comparing predictive accuracy,” *J. Bus. Econ. Statist.*, vol. 13, no. 3, pp. 253–263, 1995, doi: 10.1080/07350015.1995.10524599.

*J. Liu, J. Wang, and H. Lin, “Coordinated Physics-Informed Multi-Agent Reinforcement Learning for Risk-Aware Supply Chain Optimization,” IEEE Access, 2025, doi: 10.1109/ACCESS.2025.3629716.*

*P. Li et al., “Think4SCND: Reinforcement Learning With Thinking Model for Dynamic Supply Chain Network Design,” IEEE Access, 2024, doi: 10.1109/ACCESS.2024.3521439.*