Toward Decision-Oriented Prognostics: An Integrated Estimate-Optimize Framework for Predictive Maintenance

Zhuojun Xie, Adam Abdin, Yiping Fang*

Laboratoire Génie Industriel, CentraleSupélec, Université Paris-Saclay
xie.zhuojun, adam.abdin, yiping.fang@centralesupelec.fr

Abstract

Recent research increasingly integrates machine learning (ML) into predictive maintenance (PdM) to reduce operational and maintenance costs in data-rich operational settings. However, uncertainty due to model misspecification continues to limit widespread industrial adoption. This paper investigates a PdM framework in which sensor-driven prognostics inform decision-making under economic trade-offs within a finite decision space. We investigate two key questions: (1) Does higher predictive accuracy necessarily lead to better maintenance decisions? (2) If not, how can the impact of prediction errors on downstream maintenance decisions be mitigated? We first demonstrate that in the traditional estimate-then-optimize (ETO) framework, errors in probabilistic prediction can result in inconsistent and suboptimal maintenance decisions. To address this, we propose an integrated estimate-optimize (IEO) framework that jointly tunes predictive models while directly optimizing for maintenance outcomes. We establish theoretical finite-sample guarantees on decision consistency under standard assumptions. Specifically, we develop a stochastic perturbation gradient descent algorithm suitable for small run-to-failure datasets. Empirical evaluations on a turbofan maintenance case study show that the IEO framework reduces average maintenance regret up to 22% compared to ETO. This study provides a principled approach to managing prediction errors in data-driven PdM. By aligning prognostic model training with maintenance objectives, the IEO framework improves robustness under model misspecification and improves decision quality. The improvement is particularly pronounced when the decision-making policy is misaligned with the decision-maker's target. These findings support more reliable maintenance planning in uncertain operational environments.

Keywords: Machine Learning, Predictive Maintenance, Estimate-then-Optimize, Integrated Estimate-Optimize

1 Introduction

1.1 Background

Effective maintenance planning is essential for maximizing equipment availability, reducing operational costs, and ensuring manufacturing safety (Pintelon and Gelders, 1992). Preventive maintenance strategies have become particularly prevalent in asset-intensive industries due to their effectiveness in improving operational and asset management efficiency. For instance, organizations such as airlines (Lagos et al., 2020) and railways (Wagenaar et al., 2017) increasingly adopt preventive maintenance to enhance efficiency, minimize unplanned downtime, and streamline maintenance management (De Jonge and Scarf, 2020).

Traditionally, preventive maintenance approaches have relied on explicitly defined static models, assuming known distributions of equipment lifetimes or degradation processes, such as non-homogeneous or compound Poisson processes (Lee and Cha, 2016; Drent et al., 2023). However, these static models fail to adapt effectively under varying operational conditions, which can lead to suboptimal maintenance schedules, increased downtime, or unnecessary interventions. In contrast, data-driven predictive maintenance (PdM) leverages real-time monitoring of equipment conditions facilitated by widespread sensor deployment and advancements in data analytics methods (Wen et al., 2022). This continuous update in

---
*Corresponding author

equipment monitoring improve the accuracy of the forecast and the overall operational efficiency, which has fueled the growing interest in data-driven PdM strategies (Cai et al., 2023).

Data-driven PdM utilizes data collected from sensors to evaluate equipment health states and predict their Remaining Useful Life (RUL). This predictive process, termed prognostics (Lei et al., 2018), guides timely and cost-effective maintenance decisions. Machine Learning (ML) methods have become key enablers for PdM by capturing nonlinear, time-varying relationships between sensor data and equipment degradation. However, flexible ML models require large datasets for effective training and reliable generalization (Bartlett et al., 2021). Unfortunately, such extensive datasets are rarely available in PdM settings, as run-to-failure experiments needed to generate labeled data for supervised learning are costly and time-consuming, limiting the available sample size (Waldhauser et al., 2024). Therefore, accurately capturing the underlying relationships between sensor data and prediction targets, such as RUL, becomes difficult, making model misspecification a fundamental challenge in PdM (Lv and Liu, 2014).

Driven by recent advances in ML, predictive models have become one of the central focuses of PdM research. However, their ultimate value lies not in the prediction itself, but in supporting informed and effective maintenance decisions. In practice, these decisions are often prescribed through optimization frameworks that regard predictions of system states as input. Thus, a fundamental challenge in data-driven PdM emerges: *how can we best convert available data (e.g., sensor readings) into optimal maintenance actions?* Addressing this challenge has motivated the development of data-driven optimization, which progressively integrates ML techniques with optimization methods for effective decision-making under uncertainty. In this context, data-driven PdM stands out as a prominent application of such integration.

Data-driven optimization has increasingly incorporated ML techniques to support decision-making in the presence of *contextual information* (also known as *features* in ML and *covariates* in statistics). A prevalent methodology in this domain is the *Estimate-then-Optimize* (ETO) approach, in which optimization instances rely on predictions generated by probabilistic methods that implicitly assume high predictive accuracy (Qi et al., 2021). Most ETO studies prioritize improving estimation accuracy through increasingly sophisticated predictive models. Recent research, however, has identified a fundamental issue with ETO methods: under conditions of model misspecification, optimizing solely for prediction accuracy does not necessarily yield optimal decision quality, due to a misalignment between predictive objectives and decision goals (Elmachtoub et al., 2023; Hu et al., 2022). This misalignment was also noted by Gebraeel et al. (2023) specifically for PdM, where the decision quality is commonly measured by metrics such as maintenance costs, equipment availability, or other risk-related factors.

Motivated by these insights, this work proposes a novel data-driven PdM framework that explicitly integrates decision quality into the prognostic model learning process, establishing a unified end-to-end prognosis-then-maintenance pipeline. To our knowledge, this is the first study in PdM to directly analyze and optimize how prognostic models influence downstream maintenance outcomes. Additionally, this work provides practical insights on model comparison under misspecification, highlighting the trade-off between predictive accuracy and the quality of maintenance decisions.

1.2 Related work

This section reviews three streams of literature closely related to the decision-oriented data-driven PdM framework: (i) advances in data-driven contextual optimization, (ii) recent empirical studies in PdM using probabilistic models, and (iii) studies on decision-oriented metrics in data-driven maintenance.

1.2.1 Data-driven contextual optimization

Data-driven PdM can be viewed as an instance of data-driven contextual optimization, where sensor readings or health indicators serve as contextual information observed prior to decision-making, and the RUL is the uncertain outcome realized after decisions are made. According to a recent survey by Sadana et al. (2025), two main methodological paradigms exist for data-driven contextual optimization: *decision rules* and *learning-and-optimization*. Decision rule approaches directly map observed contextual data into feasible decisions without explicitly modeling uncertainty (Ban and Rudin, 2019; Qi et al., 2023). In contrast, learning-and-optimization approaches explicitly model conditional uncertainty—either deterministically as point estimates or probabilistically as distributions—and subsequently integrate these models into downstream optimization processes. These conditional distributions can be modeled either through nonparametric methods, such as weighted sample average approximation and its variants (Bertsimas and Kallus, 2020; Kannan et al., 2025) or through parametric approaches, such as neural networks (Mandi et al., 2024).

Recent literature in the learning-and-optimization stream emphasizes the importance of integrating predictive modeling with optimization objectives. While traditional ETO frameworks separate the prediction and optimization stages, implicitly assuming high predictive accuracy, the Integrated conditional Estimate-Optimize (IEO) approach, also known as *smart predict-then-optimize* (Elmachtoub and Grigas, 2022) or *decision-focused learning* (Mandi et al., 2024) in deterministic contexts, aligns model training directly with optimization goals.

Building upon this literature, this study introduces the first IEO framework explicitly designed for data-driven PdM. In doing so, it makes several distinct contributions to the field of data-driven optimization: (i) we extend the theoretical analysis of the IEO framework to optimization problems characterized by discrete feasible sets; (ii) we propose an efficient training algorithm to handle the non-differentiability of decision loss while preserving predictive accuracy; and (iii) we empirically investigate the practical performance of ETO and IEO approaches under varying levels of model misspecification, specifically within PdM applications.

1.2.2 Data-driven predictive maintenance

Probabilistic prognostic models have recently gained popularity in data-driven PdM due to their ability to quantify uncertainty around RUL predictions. A key advantage of probabilistic models is their compatibility with numerous maintenance strategies that rely on stochastic optimization techniques requiring distributional information. For example, Cai et al. (2023) utilized Maximum Likelihood Estimation (MLE) to estimate the failure probability distribution and, subsequently, applied a threshold-based policy for preventive maintenance. Similarly, Lee and Mitici (2023) employed a convolutional neural network combined with Monte Carlo dropout techniques (Srivastava et al., 2014) to generate probabilistic RUL estimates. They then trained a deep reinforcement learning algorithm for sequential maintenance planning using these estimates. The same neural architecture was adopted by Mitici et al. (2023) combined with single- and multi-component replacement models for scheduling maintenance actions. Other notable works include the approach by Nguyen and Medjaher (2019), who developed an LSTM-based classifier to estimate failure probabilities, integrating these predictions into a joint ordering-replacement decision model. Additionally, Zhuang et al. (2023) employed Bayesian deep learning techniques with bidirectional LSTM networks to quantify uncertainty more effectively, also for joint ordering-replacement decision-making. However, these studies predominantly adopt the ETO approach, optimizing ML models primarily for predictive accuracy. For more applications of ML in data-driven PdM, we refer to the recent review (Li et al., 2024).

While maintenance decisions resulting from these models were acceptable in specific scenarios, they have not explicitly investigated the link between predictive accuracy and decision quality. Consequently, they leave the following critical question unanswered: *does a better predictive model necessarily lead to better maintenance decisions?* To address this gap, we calibrate the prognostic models with an objective that directly minimizes the downstream maintenance cost, thereby making the trade-offs between predictive accuracy and operational decision quality explicit.

1.2.3 Decision-oriented prognostics

Recent research has underscored that evaluating prognostic models requires consideration of downstream decision-making impacts (Lewis and Groth, 2022). Particularly, Atamuradov et al. (2017) emphasized that when evaluating the performance of prognostic models, the downstream decision-making consequence and health management tasks should be taken into account. Following this direction, Kamariotis et al. (2024) proposed a decision-oriented metric to evaluate prognostic algorithms and guide hyper-parameter tuning (e.g., learning rates and step lengths in gradient descent). However, their hyper-parameter tuning heuristics offer no guarantees regarding decision quality, as multiple training processes may lead to different outcomes. We address this gap by directly incorporating decision-oriented metrics into the training process, optimizing prognostic model parameter (e.g. weights and biases for neural networks) explicitly toward improved maintenance outcomes that are effective across different decision-making models.

Finally, we differentiate our framework from a concurrent study by van der Haar et al. (2025), which also leverages ML models for feature-based maintenance with a focus on decision quality. Their approach predicts the binary *necessity* of maintenance for each available time window, thus aligning with the decision-rule approach and bypassing the need for explicit modeling of uncertainty. Consequently, their approach does not strictly fall within the scope of PdM and may have limited applicability to different maintenance problems. In contrast, our method explicitly models RUL uncertainty and thus generalizes

beyond binary decision structures. A detailed theoretical analysis demonstrating how our proposed framework encompasses and extends the approach in van der Haar et al. (2025) will be presented in Section 3.

1.3 Contributions

In summary, the contributions of this work are as follows:

1. We propose the first IEO framework for data-driven predictive maintenance, advancing decision-oriented prognostics within the PdM domain.

2. We establish a non-asymptotic generalization bound proving that the IEO framework preserves consistency for a broad class of discrete maintenance optimization problems, covering both decision-rule and learning-and-optimization settings.

3. For tractability, we develop an efficient fine-tuning scheme using a stochastic perturbation gradient method that overcomes non-differentiability in maintenance problems.

4. We validate our framework on a simulated dataset of large commercial turbofan engines, showing our framework can consistently reduce maintenance cost across diverse scenario compared to the ETO approach while generating better prognostics in certain metrics. These improvements become even more prominent as model misspecification intensifies.

2 Problem Setting

In this section, we first set up the problem by introducing RUL modeling and the representation of its uncertainty. We then formulate the downstream maintenance optimization problem based on the estimated RUL. Finally, we demonstrate the limitations of the traditional ETO approach using a simple example, thereby motivating the need for our proposed IEO framework.

2.1 RUL modeling

We consider individual equipment subject to stochastic deterioration during production or manufacturing under normal conditions. The RUL of a component, monitored via installed sensors, is specified as the health indicator. To model the uncertainty associated with its RUL, we define a discrete random variable Y, with support $\mathcal{Y}$ and realization $y \in \mathcal{Y}$. The support set is given by $\mathcal{Y} \triangleq \{\mathscr{Y}_1, \ldots, \mathscr{Y}_H\}$, where each element $\mathscr{Y}_h, h = 1, \ldots, H$, represents the aggregation of events in which the RUL of the component lies within the interval $[h, h+1)$ units of time. The definition of a time unit can be specified by the modeler to reflect the availability of maintenance resources. In this work, we define one unit of time as a single production cycle, such as one flight for a turbofan engine. The number of intervals H can be chosen by the modeler: a smaller H shortens the planning horizon but may truncate important future events, whereas a larger H extends the horizon at the cost of increased computational complexity. We let $\mathcal{P}_{\mathcal{Y}}$ be the set of all measurable probability distributions supported on $\mathcal{Y}$. Therefore, the RUL uncertainty is then represented by a probability distribution $P \in \mathcal{P}_{\mathcal{Y}}$ such that $Y \sim P$.

The contextual information (specified as sensor data in this work), which is known to have correlation with RUL, is denoted by a random variable X with its realization $x \in \mathcal{X} \subset \mathbb{R}^p$. The joint probability distribution of (X, Y) is denoted by $\mathbb{P}$. Specifically, the conditional distribution of RUL knowing the sensor data is denoted by $\mathbb{P}_{Y|X}$, hereafter referred to as *conditionals*. An estimation of the conditional $\mathbb{P}_{Y|X}$ is denoted by $\widehat{P}_{Y|X} \in \Delta_{H-1}$, where Δ_{H-1} is the $(H-1)$-dimensional probability simplex defined as $\Delta_{H-1} \triangleq \{P \in \mathbb{R}^H | P^\top \mathbf{1}_H = 1, P \geq 0\}$, and $\mathbf{1}_H$ denotes the all-one vector in $\mathbb{R}^H$. The estimation task of conditional $\widehat{P}(\cdot|X)$ can also be viewed as a probabilistic classification problem with H possible classes, based on the modeling of RUL.

2.2 Maintenance policy

In this section, we define the maintenance models considered in this work. In data-driven maintenance, the objective is to learn a policy that maps information (input) related to the sensor data x to a feasible decision z (output), in order to minimize a predefined cost or loss function. Depending on the

methodology, the input to this policy can either be the raw sensor data x, possibly after feature engineering (as in the decision-rule approach), or a statistical approximation of the conditional distribution estimated via predictive models. We adopt the latter approach since data-driven PdM fundamentally involves using observed data to infer future equipment behavior.

We now define the objective that the maintenance policy aims to optimize, which represents the operational and maintenance cost incurred by a given decision. Specifically, the cost function in Equation (1) captures the trade-off between preventive and corrective maintenance:

$$c(z; y) = \begin{cases} c_p + c_m(y - z) & \text{if } z \leq y \\ c_c + c_d(z - y) & \text{otherwise} \end{cases}, \tag{1}$$

where $z \in \mathcal{Z}$ denotes a scheduled maintenance decision in unit of cycles and $\mathcal{Z}$ its feasible region, y the random realization of RUL, c_p the preventive maintenance cost, c_c the corrective maintenance cost (larger than c_p), c_m the amortized component cost per cycle, and c_d the opportunity cost per cycle due to service interruption and unexpected shutdown (larger than c_m). Note that the stochastic realization of RUL y may not be observed after a policy has been deployed, since the recording process will stop at z if $z \leq y$. In data-driven PdM we often have offline data collected from run-to-failure experiments, we can leverage this information for model training and evaluation.

We consider two maintenance policies for the single-item preventive maintenance problem, based on *stochastic optimization* (SO) and *distributional quantile*, respectively. Both policies are designed to operate with a RUL distribution. For each policy, we assume the maintenance windows are limited to a discrete feasible set $\mathcal{Z} = \{Z_1, \ldots, Z_K\}$ to reflect the limitation of maintenance resources in reality. Here, $Z_k, \forall k = 1, \ldots, K$ denotes an available time slot for maintenance intervention. For instance, $Z_1 = 0$ represents the maintenance can be carried out at current inspection time, whereas $Z_2 = 5$ means the next available window is after 5 cycles. The feasible set $\mathcal{Z}$ is specified based on the problem setting. Next, we first assume the RUL distribution $\mathbb{P}$ is explicitly identified, thus the conditionals $\mathbb{P}_{Y|X}$, to introduce the maintenance policies. In Sections 3 and 4, we describe how to estimate RUL distributions from data.

2.2.1 Contextual stochastic optimization policy

If the RUL distribution $\mathbb{P}_Y$ is well identified, the SO problem in Equation (2) can be formulated to prescribe the optimal decision z minimizing the expected cost:

$$\textbf{(SO)} \quad z^* \in \arg\min_{z \in \mathcal{Z}} \mathbb{E}_{\mathbb{P}_Y}[c(z; Y)]. \tag{2}$$

Th SO formulation aims to find the optimal decision for cost minimization under the whole population $\mathbb{P}_Y$. If contextual information x is available–as is typically the case in PdM–and correlated with the RUL, the SO can be extended to a contextual setting, giving rise to what is known as Contextual Stochastic Optimization (CSO) (Bertsimas and Kallus, 2020; Sadana et al., 2025). In this setting, as shown in Equation (3), a CSO policy π_C tailors a decision for each scenario with observation x, rather than being optimized for the entire population:

$$\pi_C(\mathbb{P}_{Y|X=x}) \triangleq \arg\min_{z \in \mathcal{Z}} \mathbb{E}_{\mathbb{P}_{Y|X}}[c(z; Y)|X = x]. \tag{3}$$

The major difference between Equation (2) and Equation (3) is in the probability distribution over which the expectation is taken. The risk-neutral CSO model can be replaced by a variety of measures. For example, Conditional Value-at-Risk (Lin et al., 2022) and Robust Satisficing (Sim et al., 2024) have been studied in the presence of contextual information.

2.2.2 Quantile policy

In addition to CSO policy, the quantile policy–also known as the threshold policy–is widely used in the PdM literature due to its simplicity in implementation and statistical interpretability. Given sensor data x and the conditional $\mathbb{P}_{Y|X}$, the quantile policy π_Q prescribes a maintenance decision as in Equation (4):

$$\pi_Q(\mathbb{P}_{Y|X=x}) \triangleq \arg\max_{z} \left\{ z \in \mathcal{Z} : \mathbb{P}_{Y|X=x}\{Y < z\} \leq \alpha \right\}, \tag{4}$$

which selects the latest maintenance window z while ensuring that the tolerated failure probability $\alpha \in (0, 1)$ is respected. The quantile policy disconnects the decision-making and decision evaluation

process, because it disregards the cost coefficients such as c_p and c_c and focuses only on failure probability. Consequently, the tolerance α should be carefully calibrated to adapt to the downstream maintenance task for alternative targets such as cost minimization. Additionally, the risk aversion is established on the true conditional $\mathbb{P}_{Y|X}$ while in practice the probability distribution is only an estimate. Therefore, the policy may fail to meet the preset reliability requirement $1 - \alpha$.

In a data-driven PdM setting, the conditional is unknown, and the decision-maker typically relies on estimation of the RUL $\widehat{P}$. Once the estimations are available, the policies become mappings from the estimated conditionals to decisions. In the rest of the paper, we unify the notations of CSO (with subscript C) and quantile policy (with subscript Q) by $\pi(\widehat{P}; \phi)$. The dependence of decision on the sensor data x is implicitly merged into the estimated distribution $\widehat{P}$. With a slight abuse of notation, we use ϕ to denote the parameters of each policy. Specifically, for the CSO policy π_C, ϕ aggregates the cost coefficients. For the quantile policy π_Q, ϕ denotes the risk margin α.

2.3   The limitations of ETO: a motivating example

We introduce a simple example to reveal some limitations of adopting the ETO approach for PdM, particularly concerning the following assumptions: 1) *identical estimation accuracy nearly guarantees identical decision quality*, and 2) *better estimation accuracy implies better decision quality*. For illustration, we assume there exists an underlying RUL probability distribution P given sensor data x. Furthermore, we consider there exist two distinct estimation methods that estimate P based on x, each subject to inevitable estimation error δ, simulating model misspecification. The two estimations are denoted by $\widehat{P}_1$ and $\widehat{P}_2$. The underlying RUL distribution P is assumed as a truncated Poisson distribution, of which the probability mass for an integer support element k is $\frac{\lambda^k e^{-\lambda}}{k!}$ with $\lambda = 20$. The support is truncated at $H = 30$, and the probabilities are re-normalized accordingly. Cost parameters are chosen as $c_m = 1, c_d = 5, c_p = 10, c_c = 100$. The feasible decision set $\mathcal{Z} = \mathcal{Y}$ is considered for simplicity.

To quantify the estimation accuracy and decision quality of an estimated distribution, we define the two following unit-less metrics:

$$\delta_e(\widehat{P}) = \left(\frac{-\sum_{y \in \mathcal{Y}} P(y) \log \widehat{P}(y)}{-\sum_{y \in \mathcal{Y}} P(y) \log P(y)} - 1 \right) \times 100\%,$$

$$\delta_o(\widehat{P}) = \left(\frac{\mathbb{E}_P[c(\pi(\widehat{P}; \phi); Y)]}{\mathbb{E}_P[c(\pi(P; \phi); Y)]} - 1 \right) \times 100\%,$$

where $\delta_e(\widehat{P}) \geq 0$ denotes the relative estimation error (Cross Entropy) of $\widehat{P}$ to P that is consistent with the Kullback–Leibler divergence, and $\delta_o(\widehat{P}) \geq 0$ denotes the relative optimality gap incurred by the estimation $\widehat{P}$ in the expected maintenance cost. The smaller $\delta_e(\widehat{P})$, the higher accuracy in estimation; The smaller $\delta_e(\widehat{P})$, the better decision is incurred by $\widehat{P}$.

We first consider the case where two estimations $\widehat{P}_1$ and $\widehat{P}_2$ achieve near-optimal and identical estimation error $\delta_e(\widehat{P}_1) = \delta_e(\widehat{P}_2) = 1\%$. The estimated probability and the relative optimality gap are depicted in Figure 1(a) and Figure 1(b), respectively. Next, we manually set $\delta_e(\widehat{P}_1) = \delta_e(\widehat{P}_2) = 2\%$ to demonstrate the results when estimation error is larger. A similar analysis is depicted in Figure 1(c) and Figure 1(d). Clearly, in both cases, $\widehat{P}_1$ and $\widehat{P}_2$ incur different optimality gaps. Moreover, both $\widehat{P}_1$ and $\widehat{P}_2$ lead to better decision quality when the estimation error is set to 2% compared to the case in which the error is 1%. Therefore, the two assumptions indicated at the beginning of this section do not hold.

This example reveals that invariant estimation accuracy under a specific evaluation metric, such as the Kullback–Leibler divergence, can lead to distinct decision outcomes if the downstream optimization uses the estimation as input. Furthermore, lower estimation errors do not necessarily translate into lower optimality gaps under a maintenance policy. This phenomenon can be formally described by the following Proposition 1.

Proposition 1. *The relative optimality gap δ_o is not Lipschitz continuous with the estimation error δ_e. That is: given $P \in \mathcal{P}_{\mathcal{Y}}$ and two imperfect estimates $\widehat{P}_1, \widehat{P}_2 \in \mathcal{P}_{\mathcal{Y}} \backslash \{P\}$, there exists parameter values c_p, c_c, c_d, c_m, and policy π, such that $\forall C > 0$,*

$$\left| \delta_o(\widehat{P}_1) - \delta_o(\widehat{P}_2) \right| > C \left| \delta_e(\widehat{P}_1) - \delta_e(\widehat{P}_2) \right|.$$
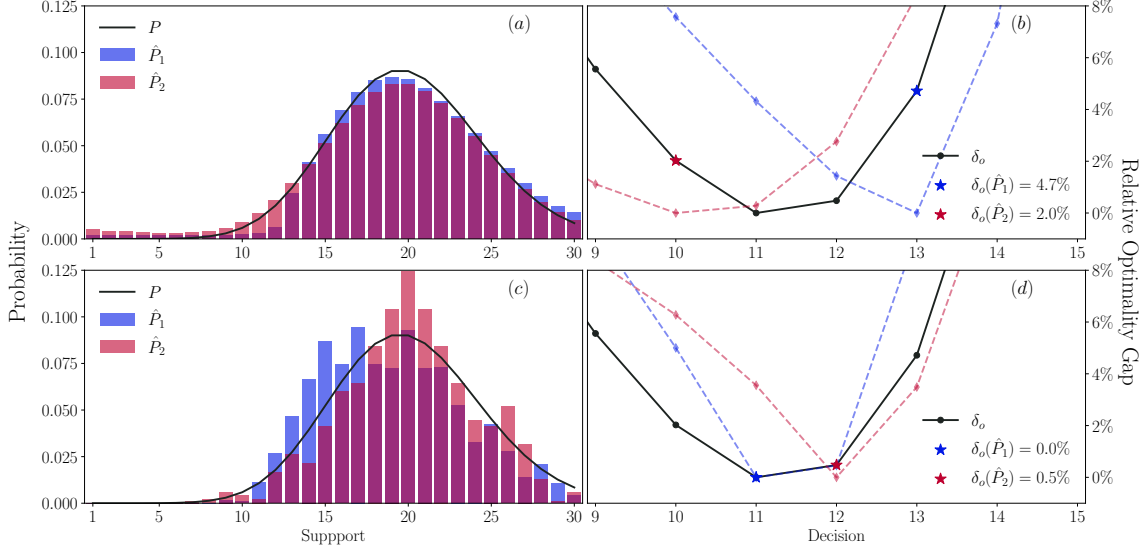
Figure 1: Estimated probability and relative optimality gap in the example. Note: (a) and (b) are for the cases where estimation error is set to $\delta_e = 1\%$, (c) and (d) are for $\delta_e = 2\%$. CSO policy is considered to prescribe decisions given $\widehat{P}_1$ and $\widehat{P}_2$. The colored dash lines in (b) and (d) indicates the objective value under the estimated distributions.

The Lipschitz condition does not hold due to the discrete nature of the feasible set $\mathcal{Z}$, which commonly arises in maintenance practice. If an ML model is chosen as the estimation method, the training configuration (i.e., hyperparameters) has been shown to have a non-negligible effect on the decision quality even if estimation accuracy remains essentially unchanged (Kamariotis et al., 2024). In the next section, we formally define the mismatch between RUL estimation and maintenance optimization within the concept of *consistency* and introduce an approach that guarantees the consistency of the learning/estimation process.

3 Consistency Analysis

In this section, we briefly revisit the ETO approach and introduce the concept of *consistency*, which characterizes the relationship between RUL estimation and maintenance optimization in PdM. We further establish a finite-sample generalization bound, providing theoretical support for the IEO approach that is consistent with the decision-making process.

3.1 Estimate-then-optimize (ETO)

In data-driven PdM, considering that RUL distribution $\mathbb{P}_{Y|X}$ is not identified, one typically relies on a dataset $\mathcal{D} \triangleq \{(x_i, y_i)\}_{i=1}^n$ containing n pairs of sensor data x and RUL labels y. The number of available samples n is assumed to be small (as is often the case in practical settings) due to limited trials of run-to-failure experiments.

In the ETO approach, when a predictive model m is employed to estimate the conditional distribution, it is typically learned by minimizing the objective defined in Equation (5), based on the dataset $\mathcal{D}$:

$$\widehat{m}_{eto} \triangleq \inf_{m \in \mathcal{M}} \frac{1}{n} \sum_{i=1}^n \ell_{eto}\left(m(x_i), y_i\right), \tag{5}$$

where $m : \mathcal{X} \to \mathcal{P}_{\mathcal{Y}}$ is a probabilistic prognostic model from a prespecified model class $\mathcal{M} \triangleq \{m(\cdot; \omega), \omega \in \Omega\}$ with Ω being a parameter space, and $\ell_{eto} : \mathcal{P}_{\mathcal{Y}} \times \mathcal{Y} \to \mathbb{R}$ denotes a chosen loss function. In this work, we consider the Negative Log-Likelihood (NLL) as the loss function, which is defined in Equation (6):

$$\ell_{nll}(m(x), y) = -\log m(y|x). \tag{6}$$

With a slight abuse of notation, $m(y|x)$ here denotes the probability mass the model m assigns to the element $y \in \mathcal{Y}$ given input x, with parameter ω omitted.

The variational problem in Equation (5) is usually replaced in ML studies by a regularized, parametric surrogate:

$$\widehat{\omega}_{eto} \triangleq \min_{\omega \in \Omega} \sum_{i=1}^{n} \ell_{nll} \left(m(x_i; \omega), y_i \right) + \lambda \left\| \omega \right\|_2, \tag{7}$$

where $\lambda \geq 0$ denotes the strength of the ℓ_2 regularization. The optimizer $\widehat{\omega}_{eto}$ is non-trivial to obtain because the mapping $\omega \mapsto \sum_{i=1}^{n} \ell_{nll}(m(x_i; \omega), y_i) + \lambda \|\omega\|_2$ is often nonconvex in ω. Stochastic Gradient Descent (SGD) and its variants, such as Adam optimizer (Kingma and Ba, 2014), are commonly used in ML to reach near-optimal parameter, often in combination with learning strategies such as dropout (Srivastava et al., 2014). Once the optimal or near-optimal model parameter $\widehat{\omega}_{eto}$ has been obtained, the fitted model is used in the decision-making process to estimate the RUL distribution for a new unseen instance x_0. The estimation $m(x_0; \widehat{\omega}_{eto})$ is then passed to a policy $\pi(\cdot; \phi)$ to generate a maintenance decision. However, as illustrated in the examples in Section 2, the combination of a predictive model and a downstream policy can yield counterintuitive or suboptimal outcomes. Recent theoretical developments further highlight the limitations of the ETO approach, particularly for decision-making under considerable model misspecification (Hu et al., 2022; Elmachtoub et al., 2023).

3.2 Definition of consistency

To understand the link between RUL estimation accuracy and maintenance consequences, we contrast two notions of risk. *Decision risk* $R(\omega)$, also referred to as true risk, is the expected maintenance cost incurred by the model parameter ω when applying the model $m(\cdot; \omega)$ to the predictive maintenance task with a fixed policy π under objective c:

$$R(\omega) \triangleq \mathbb{E}_{\mathbb{P}}[c(\pi(m(X; \omega); \phi); Y)]. \tag{8}$$

Meanwhile, *estimation risk* $R_\ell(\omega)$, also known as surrogate risk, measures how well the model predicts independently of any decision-making policy, under a user-chosen loss ℓ:

$$R_\ell(\omega) \triangleq \mathbb{E}_{\mathbb{P}}[\ell(m(X; \omega), Y)]. \tag{9}$$

Specifically, the objective function in Equation (5), with $\lambda = 0$ corresponds to the empirical counterpart of estimation risk in Equation (9). Under standard assumptions, the empirical minimizer $\widehat{\omega}_{eto}$, depending on the model complexity and the size of the training set, guarantees the estimation risk (Bartlett et al., 2021). However, the guarantee on estimation risk cannot directly translate into guarantees on decision risk. In the context of PdM, this means that improvements in the estimation accuracy of the model m, as measured by the loss function ℓ on the data set $\mathcal{D}$, do not necessarily imply a reduction in the overall maintenance cost resulting from the joint deployment of m and π. As discussed in Section 2, this disconnect arises because the maintenance cost function lacks a robust Lipschitz property due to the discreteness of decision spaces, rendering the cost function sensitive to even small estimation errors.

We next proceed to formally define the concept of *model misspecification*.

Definition 1 (model misspecification). *A model class $\mathcal{M}$ is misspecified w.r.t. $\mathbb{P}$ if*

$$\forall m \in \mathcal{M}, \int_{\mathcal{X}} \mathbb{D}_{KL} \left(\mathbb{P}_{Y|X=x} || m(x) \right) d\mathbb{P}_X(x) > 0.$$

Contrarily, a model class $\mathcal{M}$ is well-specified w.r.t. $\mathbb{P}$ if

$$\exists m \in \mathcal{M}, \int_{\mathcal{X}} \mathbb{D}_{KL} \left(\mathbb{P}_{Y|X=x} || m(x) \right) d\mathbb{P}_X(x) = 0.$$

where $\mathbb{D}(\cdot||\cdot)$ denotes the Kullback–Leibler divergence of two distributions over the same support.

Model misspecification is an unavoidable aspect of data-driven methodologies. Rather than assuming access to a well-specified model class, whose validity typically requires large sample sizes to verify, it is more critical to understand how misspecification influences decision-making when the learning objective is to identify a best-in-class, accuracy-oriented estimator under limited data. To rigorously characterize this relationship, we adopt the notion of *consistency*, as introduced in Ho-Nguyen and Kılınç-Karzan (2022), which formalizes the extent to which improvements in predictive performance translate into better decision outcomes.

Definition 2 (ℙ-consistency). *A loss function ℓ is ℙ-consistent, or with ℙ-consistency, if for all $\epsilon > 0$, there exists a $\gamma > 0$ (which may depend on ϵ), such that:*

$$\left\{\omega \big| R_\ell(\omega) - \inf_{\omega' \in \Omega} R_\ell(\omega') \le \gamma\right\} \subseteq \left\{\omega \big| R(\omega) - \inf_{\omega' \in \Omega} R(\omega') \le \epsilon\right\}.$$

The ℙ-consistency condition describes how a loss ℓ supports decision making for a specified model class $\mathcal{M}$, policy π, and data distribution ℙ. If ℓ is ℙ-consistent, then minimizing the estimation risk—thereby obtaining the best predictive model in $\mathcal{M}$—yields a decision risk no worse than that of the best decision-oriented model in the same model class, up to a certain level. For certain policy classes, Ho-Nguyen and Kılınç-Karzan (2022) showed that ℙ-Fisher consistency of ℓ is sufficient to guarantee ℙ-consistency. Nonetheless, establishing ℙ-consistency for general losses is challenging because it hinges on the specific structure of the policy π and the downstream optimization problem.

Despite the difficult of proving the consistency of a surrogate loss, there at least exists a ℙ-consistent loss L, defined in Equation (10), which replaces the estimation error by the decision loss evaluated by plugging decision $\pi(m(X); \phi)$ into the underlying cost structure c:

$$L(m(X), Y) \triangleq c(\pi(m(X); \phi); Y). \tag{10}$$

We show the loss L holds the ℙ-consistency by the following Proposition 2.

Proposition 2. *The loss function L in Equation (10) is ℙ-consistent for any model class $\mathcal{M}$, since by definition $R_L(\omega) = R(\omega)$. Specifically, $\forall \gamma > 0$, there always exist an $\epsilon \ge \gamma$, such that:*

$$\left\{\omega \big| R_L(\omega) - \inf_{\omega' \in \Omega} R_L(\omega') \le \gamma\right\} \subseteq \left\{\omega \big| R(\omega) - \inf_{\omega' \in \Omega} R(\omega') \le \epsilon\right\}.$$

3.3 Non-asymptotic guarantee of consistency

A critical challenge remains in minimizing the risk $R_L(\omega)$. Proposition 2 establishes that minimizing the *estimation risk* of the parameter ω under L is equivalent to minimizing the true risk, but it does not suggest any approach to minimize the risk. Since the true distribution ℙ is unknown, models must be trained using a finite dataset $\mathcal{D}$. This introduces a key concern: while $R_L(\omega) = R(\omega)$ in the population limit due to the consistency of L, how can we ensure that the parameters ω learned from empirical data $\mathcal{D}$ approximates a minimizer of $R_L(\omega)$? To address this gap, we leverage the framework of Probably Approximately Correct (PAC) learning to derive probabilistic guarantees for model parameter ω that minimizes the empirical risk over $\mathcal{D}$. The following outlines the key assumptions and components required to establish such a generalization bound.

Assumption 1. *The dataset $\mathcal{D}$ is i.i.d. generated from a data-generating process $\mathbb{P}^n \triangleq \frac{1}{n}\sum_{i \in [n]} \delta_{(X_i, Y_i)}$ with $\delta_{(\cdot, \cdot)}$ the Dirac measure, and $(X_i, Y_i) \sim \mathbb{P}$ for all $i = 1, \ldots, n$.*

Assumption 2. *The maintenance cost function is bounded, i.e., $\forall z \in \mathcal{Z}, \forall y \in \mathcal{Y}, |c(z; y)| \le C_1$ for some $C_1 > 0$.*

Assumption 3. *The feasible set $\mathcal{Z}$ is finite, i.e., $|\mathcal{Z}| = K$ with $K < \infty$.*

Next, we unify the RUL estimation and maintenance decision-making into an integrated policy. This procedure has two advantages. First, it allows our consistency guarantee to apply to both the decision-rule and the learning-and-optimization approaches. Second, it enables us to leverage the Natarajan dimension, which is defined in Definition 4, of the integrated policy to construct an upper bound on the model complexity term involved in the PAC-type guarantee.

Definition 3 (Integrated Policy Class). *A model class $\mathcal{G}$ denotes an end-to-end function class contains mappings from contextual information to finitely possible decisions:*

$$\mathcal{G} \triangleq \{g_\omega : \mathcal{X} \to \mathcal{Z}, \omega \in \Omega\}.$$

The predictive maintenance process can be considered as $g_\omega(x) \triangleq \pi(m(x; \omega); \phi)$. Furthermore, the class can be generalized to a decision-rule approach, such as one where a model $g(\cdot; \omega)$ generates weights (analogous to action selection in reinforcement learning) over all possible decisions in $\mathcal{Z}$. When binary weights are generated for each $z \in \mathcal{Z}$ by model g_ω, the class contains the policy proposed by van der Haar et al. (2025).

With the above elements established, we derive a theoretical guarantee for our framework in Theorem 1. Our analysis is inspired by the work of El Balghiti et al. (2023), who utilize the Natarajan dimension to establish performance guarantee for the SPO loss (Elmachtoub and Grigas, 2022). Their approach interprets linear programs with polyhedral constraints as discrete choice problems over a finite set of vertices, a perspective that informs our treatment of discrete decision spaces in PdM. Note that our theoretical development assumes the Natarajan dimension of the model class $\mathcal{G}$, denoted by d, is smaller than n.

Theorem 1. *Suppose Assumptions 1-3 hold. If the loss ℓ in Equation (7) is replaced by L with $\lambda = 0$, and the Natarajan dimension d is less than or equal to n, then with probability at least $1 - \delta$ over the draw of $\mathcal{D}$ from $\mathbb{P}^n$, the following inequality holds for all $\omega \in \Omega$:*

$$R_L(\omega) \leq \widehat{R}_L(\omega) + C_1 \sqrt{\frac{\log(1/\delta)}{2n}} + 2C_1 \sqrt{\frac{2d\log(en/d) + 4d\log K}{n}}$$

where $\widehat{R}_L(\omega)$ denotes the empirical estimation risk under L, obtained by replacing ℓ in Equation (7) with L as defined in Equation (10).

The proof can be found in the Appendix A. Theorem 1 states the probabilistic guarantee on the true risk $R_L(\omega)$ of model parameter ω, given its empirical risk $\widehat{R}_L(\omega)$, regardless of how ω is determined. The following corollary, of which the proof is provided in the Appendix A, further provides the consistency of the learning process, showing that a model learned by minimizing $\widehat{R}_L(\omega)$ has a high probability to approach the best-in-class decision-oriented model. Thus, the consistency of the learning process is guaranteed.

Corollary 1. *Denote one minimizer of $\widehat{R}_L(\omega)$ as ω_{erm}. Suppose Assumptions 1-3 hold and $d \leq n$. Then, the following inequality holds with probability at least $1 - \delta$ over the draw of $\mathcal{D}$ from $\mathbb{P}^n$:*

$$R_L(\omega_{erm}) \leq \min_{\omega \in \Omega} R_L(\omega) + 2C_1 \sqrt{\frac{\log(2/\delta)}{2n}} + 2C_1 \sqrt{\frac{2d\log(en/d) + 4d\log K}{n}}$$

Corollary 1 provides a probabilistic guarantee of $\mathbb{P}$-consistency for the empirical minimizer ω_{erm}, which is practical (though not trivial) to obtain given the dataset $\mathcal{D}$, showing convergence towards the best-in-class true risk minimizer. As long as ω_{erm} can be obtained for a dataset $\mathcal{D}$, the excess risk, defined as $R_L(\omega_{erm}) - \min_{\omega \in \Omega} R_L(\omega)$, is bounded in probability. As $n \to \infty$, the excess risk converges to 0 with a rate $\mathcal{O}(\sqrt{\log n/n})$, meaning that the learning process of ω by minimizing the empirical risk is consistent as minimizing its excess risk and, therefore, the true risk.

Remark 1 (Dependent and Limited data)**.** *To establish the results based on McDiarmid's inequality, we adopt the i.i.d. assumption. In practical maintenance applications, sensor data are typically time-series and are often preprocessed using sliding windows, where successive samples may share overlapping information. Consequently, the i.i.d. assumption might not necessarily hold. We refer to Bertsimas and Kallus (2020, Theorem EC.10) for a generalization result of certain mixing processes. Additionally, Theorem 1 establishes a uniform generalization bound that holds $\forall \omega \in \Omega$. Consequently, the bound becomes informative primarily in the regime where the sample size n is large. In data-driven PdM, the available dataset is typically small. In this setting, the bound may be dominated by the two error terms, thereby limiting its practical tightness.*

The Remark motivates real-world evaluation of the IEO framework in data-driven PdM, moving beyond purely theoretical analysis. With the consistency of the loss function L established, our focus now shifts to empirical risk minimization. In Section 4, we propose a practical model training framework. Before that, we introduce empirical evaluation metrics of practical relevance.

3.4 Empirical evaluation metrics

With no direct access to the real distribution $\mathbb{P}$, we consider an independent testing dataset $\mathcal{D}_t$ separate from the training set $\mathcal{D}$. The evaluation metrics described below are computed using this testing dataset. Four empirical evaluation metrics, namely: (i) regret, (ii) failure frequency, (iii) NLL loss, and (iv) Mean Absolute Error (MAE) loss are considered to evaluate the performance of PdM

frameworks, which are defined as:

$$\frac{1}{|\mathcal{D}_t|} \sum_{(x_i,y_i)\in\mathcal{D}_t} c(\pi(m(x_i;\omega);\phi);y_i) - \min_{z\in\mathcal{Z}} c(z;y_i), \tag{i}$$

$$\frac{1}{|\mathcal{D}_t|} \sum_{(x_i,y_i)\in\mathcal{D}_t} \mathbb{I}[z_i > y_i], \tag{ii}$$

$$\frac{1}{|\mathcal{D}_t|} \sum_{(x_i,y_i)\in\mathcal{D}_t} -\log(m(y_i|x_i,\omega)), \tag{iii}$$

$$\frac{1}{|\mathcal{D}_t|} \sum_{(x_i,y_i)\in\mathcal{D}_t} \left| \arg\max_{\mathscr{Y}\in\mathcal{Y}} m(\mathscr{Y}|x_i,\omega) - y_i \right|. \tag{iv}$$

Note in the CSO policy, high failure frequency is implicitly penalized through high cost coefficients c_d and c_c, whereas the quantile policy explicitly constrains the probability of failure to be at most α for each individual instance.

4 Integrated Estimate-Optimize (IEO) Framework

In this section, we propose a practical IEO framework for data-driven PdM. The framework integrates with an RUL prediction model formulated as a neural network-based high-dimensional classifier over a discrete support, combined with either the CSO or quantile-based policy. To reduce complexity, the predictive component adopts a Weibull-type parametrization. To address the computational challenges of IEO, we introduce a fine-tuning strategy along with an efficient stochastic perturbation algorithm to improve decision quality.

4.1 RUL predictive model

We consider the RUL distributions are represented by a vector of probabilities over the discrete support $\mathcal{Y}$. Therefore, the predictive model class $\mathcal{M}$ should be capable of generating a probabilistic prediction potentially in high-dimensional space $\mathcal{P}_{\mathcal{Y}}$ depending on the choice of H. Various probabilistic prognostics algorithms have been investigated in PdM to estimate RUL conditioned on sensor observations (see Section 1). To focus on the consistency of frameworks instead of dataset-specific model selection and hyperparameter tuning, we adopt a Weibull-type representation of the RUL distribution for its computational tractability. The Weibull distribution is widely used in reliability engineering for lifetime estimation (Canavos and Taokas, 1973). Compared to the discrete Poisson distribution, the two-parameter Weibull model offers greater flexibility.

A Weibull-type RUL distribution is generated as follows: First, the predictive model does not output an H-dimensional distribution directly, but instead produces two parameters $m(x;\omega) = \theta(x) = [\lambda(x), k(x)]$ denoting the aggregation of scale and shape parameters for the Weibull distribution $\mathcal{W}(k,\lambda)$. Next, the continuous Weibull distribution $\mathcal{W}(k,\lambda)$ is projected onto $\mathcal{Y}$ to obtain the predicted conditional $\widehat{P}(\cdot|\theta(x))$:

$$\tilde{P}(y|\theta(x)) = \frac{k(x)}{\lambda(x)} \left(\frac{y}{\lambda(x)}\right)^{k(x)-1} e^{-(y/\lambda(x))^{k(x)}},$$

$$\widehat{P}(y|\theta(x)) = \frac{\tilde{P}(y|\theta(x))}{\sum_{y'\in\mathcal{Y}} \tilde{P}(y'|\theta(x))}, \quad \forall y \in \mathcal{Y}.$$

The process first truncates then re-normalizes the probability density of $\mathcal{W}(k,\lambda)$ over $\mathcal{Y}$ to obtain a Weibull-type probability vector. It is important to emphasize that the use of a Weibull-type representation in our predictive model does not imply an assumption that the true conditional distribution $\mathbb{P}_{Y|X}$ belongs to the Weibull family. In traditional settings, modelers often impose such distributional assumptions explicitly by specifying a likelihood function in MLE or Bayesian inference. Our framework does not rely on likelihood, but instead leverages the Weibull representation to generate a discrete distribution, which is evaluated either by the NLL loss of probability (in the ETO approach) or the decision quality (in the IEO approach). The Weibull representation reduces the output dimension from H to 2 while maintaining a considerable degree of model flexibility. However, it cannot capture potential multimodality in the RUL distribution, which may occur when multiple fault modes exist. This limitation

can be addressed by employing a mixture of Weibull. Other probabilistic generative methods, such as normalizing flow (Papamakarios et al., 2021), also offer promising alternatives and are left for future research.

4.2 Framework overview

We present the IEO framework for data-driven predictive maintenance, which combines a predictive model with a preselected policy. Building on Theorem 1 and Corollary 1, we formulate an empirical decision risk minimization problem in Equation (11) that fine-tunes a model trained using the ETO approach:

$$\widehat{\omega}_{ieo} = \operatorname*{arg\,min}_{\omega \in \Omega(\widehat{\omega}_{eto})} \frac{1}{n} \sum_{i=1}^{n} L(m(x_i; \omega), y_i), \tag{11}$$

where the parameter space $\Omega(\widehat{\omega}_{eto})$ is required to be proximal to the ETO-trained model parameter $\widehat{\omega}_{eto}$. We denote the fine-tuning region $\Omega(\widehat{\omega}_{eto})$ as a parameter space centered at $\widehat{\omega}_{eto}$ with specific constraints. It could be, for instance, $\{\omega \in \Omega \,|\, \|\omega - \widehat{\omega}_{eto}\|_2 \leq r\}$ with a tunable radius parameter r. In this work, we achieve it by limiting the step size and the number of training steps of the fine-tuning process. The fine-tuning approach mainly reduces the computational expense of decision-oriented learning while preserving the predictive accuracy by searching the parameters in the neighborhood of $\widehat{\omega}_{eto}$. Alternative training strategies based on regularization (either on parameters or predictive accuracy) exist, see Qi et al. (2021); Stratigakos et al. (2024).

4.3 Stochastic perturbation gradient descent

One of the major challenges in the IEO framework is to obtain the gradient of L with respect to the model parameter ω. By simplifying the notation as $m(x_i; \omega) = \theta(x_i) = \theta_i$ and $L(\theta_i, y_i) = L_i$ for each sample (x_i, y_i), one can leverage the chain rule to obtain the gradient:

$$\nabla_\omega L_i = \frac{\partial L_i}{\partial \theta_i} \frac{\partial \theta_i}{\partial \omega} = \frac{\partial L_i}{\partial \pi(\widehat{P}(\cdot|\theta_i); \phi)} \frac{\partial \pi(\widehat{P}(\cdot|\theta_i); \phi)}{\partial \theta_i} \frac{\partial \theta_i}{\partial \omega}$$

where $\partial \theta_i / \partial \omega$ denotes the prediction gradient with respect to the model parameter, which can be easily computed using existing tools such as automatic differentiation in *PyTorch* (Paszke et al., 2019). The term $\partial L_i / \partial \theta_i$ quantifies how the decision quality L_i varies locally with respect to the model output at θ_i, and $\pi(\widehat{P}(\cdot|\theta_i); \phi)$ concisely represents the process of constructing a Weibull-type discrete distribution $\widehat{P}(\cdot|\theta_i)$ and embedding it into the policy π for decision-making.

Two computational issues arise when deriving the analytical gradient. First, the gradient of decision quality with respect to the decision can be identically zero in certain maintenance problems (e.g., when $c_d = c_m = 0$). Second, the policy mapping from the predicted distribution to the decisions is non-continuous due to the discrete nature of $\mathcal{Z}$, which makes the gradient of the decision with respect to the prediction undefined.

To circumvent these computational challenges, we develop a stochastic perturbation algorithm which approximates the gradient $\nabla_\theta L$ by bridging θ to L directly. Consider a two-dimensional standard Gaussian noise $\eta \sim \mathcal{N}(0, I)$, where $I \in \mathbb{R}^{2 \times 2}$ is the identity matrix, and let Σ be a positive semi-definite symmetric matrix that defines the covariance. For a given perturbation η, the noisy prediction $\theta_i + \Sigma \eta \sim \mathcal{N}(\theta_i, \Sigma \Sigma^\top)$ may lead to a different decision than the original θ_i (though not always). Consequently, this results in a different decision loss being observed, thereby alleviating the issue of vanishing gradients.

Next, a smooth approximation of L_i can be constructed by taking the expectation over η:

$$\tilde{L}_i \triangleq \mathbb{E}_\eta[L(\theta_i + \Sigma \eta, y_i)].$$

The gradient of the smoothed loss $\tilde{L}_i$ with respect to the model output θ_i admits an explicit formulation by considering the Gaussian random variable $\tilde{\theta}_i = \theta_i + \Sigma \eta$, whose probability density function is parametrized by θ_i. To obtain a smooth gradient without differentiating through the policy, we employ the score function gradient estimation (Mohamed et al., 2020):

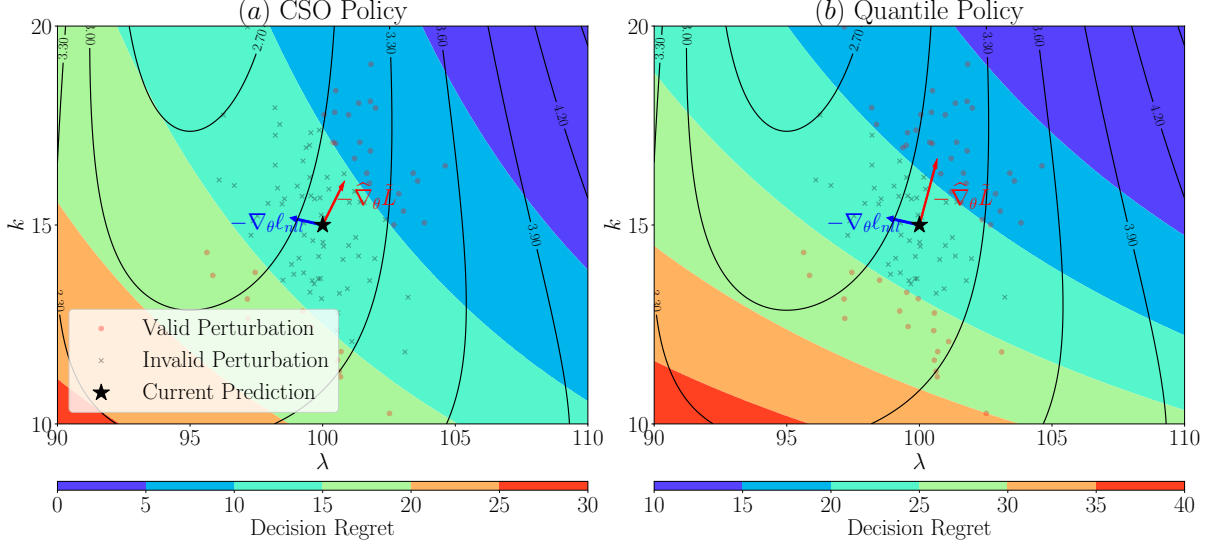$$\nabla_\theta \tilde{L}_i = \Sigma^{-1} \mathbb{E}_\eta[L(\theta_i + \Sigma \eta, y_i)\eta]. \tag{12}$$

Figure 2: Gradients generated by the ETO and the IEO approaches. *Note.* Gradients for NLL loss reduction $-\nabla_\theta \ell_{nll}$ and for regret minimization $-\widehat{\nabla}_\theta \tilde{L}$ over the landscape of regret for current prediction $\theta = [\lambda, k] = [100, 15]$ while the label is $y_i = 95$. We set $M = 100$ and $\Sigma = I$. Solid black line denotes the contour of NLL loss. The red dots denote valid perturbations that have different regret compared to current prediction, while black crosses denote invalid perturbations. CSO policy is combined in (a) and quantile policy in (b).

Practically, the expectation in Equation (12) can be approximated by Monte-Carlo sampling using M i.i.d. samples $\{\eta^j\}_{j=1}^M$ drawn from $\mathcal{N}(0, I)$, resulting in an unbiased estimator for $\nabla_\theta \tilde{L}_i$ (Berthet et al., 2020):

$$\widehat{\nabla}_\theta \tilde{L}_i \approx \frac{\Sigma^{-1}}{M} \sum_{j=1}^M L(\boldsymbol{\theta} + \Sigma \eta^j, y) \eta^j. \tag{13}$$

The advantage of Equation (13) lies in its ability to decouple gradient estimation from the need to differentiate through the optimization process, requiring only forward computations (M times per sample). This decoupling is particularly beneficial when the downstream optimization is discrete–as is commonly the case in maintenance settings–where analytical gradients typically do not exist.

However, in practice, the estimated gradient $\widehat{\nabla}_\theta \tilde{L}_i$ may suffer from high variance. To address this, we adopt the REINFORCE (Williams, 1992), introducing a baseline $b(\theta_i) = L(\theta_i, y_i)$ to reduce sampling variance and avoid the need for a large number of samples M. Other approaches for variance reduction exist, among which we conjecture an importance sampling technique for noise η dedicated to the downstream task may be a promising alternative.

Figure 2 illustrates the negative gradient $-\widehat{\nabla}_\theta \tilde{L}$ over the landscape of decision quality, expressed in terms of regret. The regret landscape reveals that there exist infinitely many estimations (λ, k) that yield the same decision (and thus the same regret), but differ in estimation accuracy, and vice versa. Furthermore, the gradient for minimizing the NLL loss, denoted by $-\nabla_\theta \ell_{nll}$ does not align with the gradient $-\widehat{\nabla}_\theta \tilde{L}$ for cost minimization. This discrepancy holds true for both policies considered in our study.

5   Case Studies on Turbofan Engine Maintenance

5.1   Experimental setup

We conduct numerical experiments using simulated data from a realistic large commercial turbofan engine, based on the CMAPSS dataset provided by NASA (Saxena et al., 2008). The dataset consists of four subsets, labeled *FD001* through *FD004*. The *FD001* subset was generated under a single operational condition with one fault mode. In contrast, *FD002* and *FD004* include six operational conditions, while *FD003* and *FD004* contain two fault modes. Our study focuses on the *FD001* subset for two main reasons: (1) In operations and maintenance, handling multiple fault modes typically requires distinct

Table 1: Model training configuration used for all experiments. *Note. C* and *Q* denotes the combined CSO and quantile policy, respectively. The ETO framework has no dependency on the policy thus the policy is omitted.

Framework	Optimizer	Learning Rate	Dropout Rate	Batch Size	Steps
ETO	Adam	1e-3	10%	64	200 + 100
IEO-C	Adam	2e-4	10%	64	100
IEO-Q	Adam	2e-4	10%	64	100

prognostics and tailored maintenance strategies. (2) More importantly, multiple operational conditions introduce non-identical data distributions. In such scenarios, decision-makers must account for the decision-dependent uncertainty to balance revenue and maintenance costs (Basciftci et al., 2020; Drent et al., 2024). These complexities fall outside the scope of our current study, which concentrates on single-component maintenance and aims to highlight the structural differences between the ETO and IEO frameworks.

The *FD001* subset includes both a training set and a test set. The training set contains run-to-failure data from 100 engines of the same type. The test set also includes data from 100 engines; however, the recordings stop before failure, with the corresponding true RUL provided for each case. For example, the first engine in the test set has sensor data recorded up to 112 cycles before failure. This introduces a distributional shift between the training and testing data due to incomplete sequences. To ensure consistent evaluation over the full operational lifetime, we restrict our analysis to the engines in the training set, which contain full run-to-failure sequences. Our feature engineering follows Mitici et al. (2023), employing a sliding window of size 30 to extract time-series sensor features. This results in a feature space of dimension $\dim(\mathcal{X}) = 30 \times 14 = 420$. A sliding-window approach with unit stride is used to generate multiple training instances per engine.

We evaluate four framework variants: ETO-C, ETO-C, IEO-C, and IEO-Q. Here, ETO and IEO refer to different frameworks, while C and Q denote the combined CSO and quantile policy, respectively. To focus on the structural differences between frameworks, we use a simple neural network architecture with two hidden layers of dimensions (400, 100), mapping the 420-dimensional input to 2 outputs to conform to our Weibull-type representation. Each hidden layer incorporates a 10% dropout rate to mitigate overfitting. To simulate a fine-tuning process, we first train the model using the ETO strategy for 200 steps. The resulting model parameter is then transferred to the IEO framework for an additional 100 steps, using a reduced learning rate. For a fair comparison, the ETO framework continues training for additional 100 steps. Table 1 lists the details of the training configuration. Other parameters are determined as follows: $H = 150, \mathcal{Z} = \{0, 5, \ldots, 125\}$. For the underlying maintenance cost function and CSO policy, $c_m = 1, c_d = 5, c_p = 50, c_c = 200$ are considered. For the quantile policy, we use $\Sigma = I, M = 1000, \alpha = 0.01$.

5.2 Base case: In-sample evaluation without distributional shift

To provide a clear comparison between the ETO and IEO frameworks, we begin by designing an ideal scenario without any distributional shift between the training and testing datasets. Specifically, we use all 100 engines for both training and evaluation, directly assessing performance on the training set. This case study enables us to examine whether improved predictive accuracy translates to improved decision quality at a population level. We include only data pairs with RUL labels of 125 or less to align with the latest maintenance window (set as 125), also to control the dataset size to manage the degree of model misspecification. To ensure robustness, the experiments are independently repeated 100 times. The results of these experiments are illustrated in Figure 3. Table 2 further lists the performance of frameworks under various metrics.

First, our results demonstrate that the model fine-tuned via IEO approach significantly improves decision quality in terms of regret. Specifically, IEO reduces the average regret by approximately 4.3% compared to the ETO framework with the CSO policy and by 27.8% with the quantile policy. This improvement in decision quality comes with a modest trade-off—approximately a 6% increase—in the NLL loss, which is the learning objective of the ETO framework. More notably, the IEO-C and IEO-Q frameworks substantially reduce variability in regret across the 100 trials. Table 2 quantifies this variability by the standard deviation and maximum value of regret across 100 trials. The maximum regret is reduced by 12.9% and 34.1% in IEO-C and IEO-Q, respectively.

The decision quality improvement by employing the IEO approach is particularly notable combining
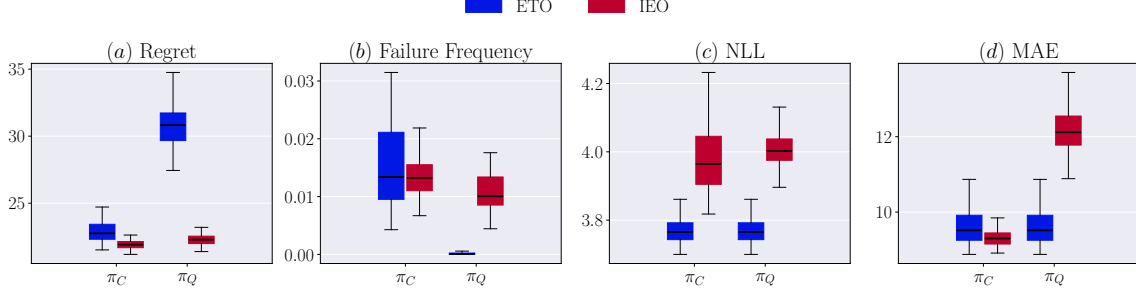
Figure 3: Performance of four frameworks under different evaluation metrics in the ideal case. *Note.* The boxes are plotted with the 25% and the 75% confidence intervals. The ideal case evaluates the metrics over the training set of 100 engines.

Table 2: Evaluation metrics of four frameworks in the ideal case by 100 experiments. *Note.* Evaluation metrics are demonstrated in mean value $\pm$ the standard deviation considering 100 independent experiments.

Framework	Regret	Max Regret	Failure Frequency	NLL	MAE
ETO-C	22.918 ± 0.869	26.223	0.016 ± 0.009	$\mathbf{3.772} \pm 0.043$	9.629 ± 0.534
IEO-C	$\mathbf{21.930} \pm 0.326$	$\mathbf{22.850}$	0.013 ± 0.003	3.995 ± 0.126	$\mathbf{9.314} \pm 0.205$
ETO-Q	30.866 ± 1.699	35.952	$\mathbf{0.000} \pm 0.000$	$\mathbf{3.772} \pm 0.043$	9.629 ± 0.534
IEO-Q	22.277 ± 0.367	23.691	0.011 ± 0.003	4.009 ± 0.051	12.183 ± 0.642

the quantile policy. The ETO-Q framework exhibits over conservativeness in decision-making, with an average regret of 30.866 and zero failure occurrences. Conversely, the IEO-Q framework achieves a well-calibrated failure frequency, closely matching the preset risk margin of 1%, while simultaneously outperforming the ETO frameworks in terms of regret. Additionally, we note that the MAE loss for the IEO-C framework, i.e. 9.314, is the lowest among all methods with the smallest variability.

This case study highlights the inherent inconsistency between the NLL loss and the actual goals of downstream tasks, such as minimizing maintenance costs and reducing failures. The IEO approach consistently improves decision quality and maintains stable performance when integrated with both the CSO and quantile policies. Notably, the value of the IEO framework becomes particularly pronounced when the utilized decision policy is less aligned with the underlying objective in the downstream maintenance problem.

5.3 Short-term scheduling: Out-of-sample evaluation under weak misspecification

In the second case study, we consider a more practical setting by evaluating framework performance on out-of-sample data. Consistent with our previous setup, we focus exclusively on data pairs with RUL labels of 125 or less. This approach allows us to control the degree of model misspecification and simulate realistic short-term scheduling scenarios in PdM applications, as if extra information indicates the need for maintenance. Specifically, we use data from the first 20 engines for testing and the remaining 80 engines for training. To ensure robust and reliable results, the experiments are repeated independently 100 times. Results are illustrated in Figure 4 and summarized numerically in Table 3.

The results again confirm that the IEO approach consistently reduces both the average and the variability of regret when paired with either the CSO or quantile policies. Notably, the IEO-Q framework demonstrates the best performance across decision-related metrics compared to all other alternatives, including IEO-C, albeit with a larger trade-off in terms of NLL and MAE losses. Similar to the first case study, IEO-C achieves the lowest MAE loss overall, though it slightly increases the failure frequency from 0.02 to 0.023 relative to ETO-C. The risk calibration of the IEO-Q framework is somewhat affected by the distributional shift between training and testing datasets, yet it still maintains the highest level of safety in decision-making, second only to the over-conservative ETO-Q framework. Overall, consistent with earlier findings, the IEO-Q framework delivers competitive decision-making performance compared to the IEO-C framework when the underlying maintenance objective is not explicitly conveyed to the quantile policy.
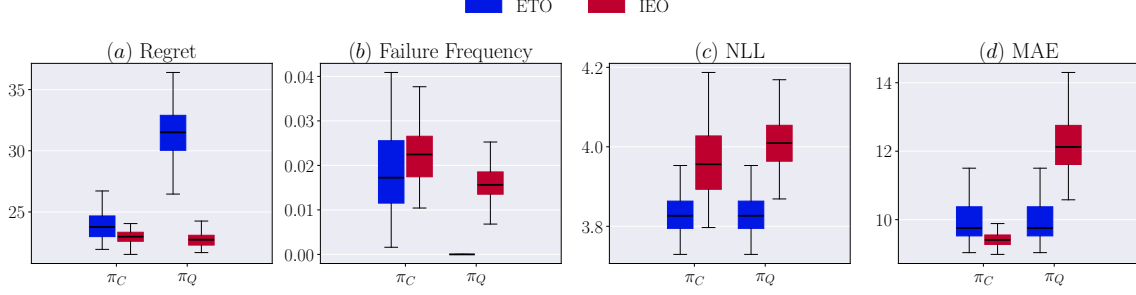
Figure 4: Performance of four frameworks under different evaluation metrics in the short-term scheduling case by 100 experiments. *Note.* The boxes are plotted with the 25% and the 75% confidence intervals. The short-term scheduling case evaluates the metrics over 20 testing engines.

Table 3: Evaluation metrics of four frameworks in the short-term scheduling case. *Note.* Evaluation metrics are demonstrated in mean value $\pm$ the standard deviation considering 100 independent experiments.

Framework	Regret	Max Regret	Failure Frequency	NLL	MAE
ETO-C	24.026 ± 1.322	27.812	0.020 ± 0.011	$\mathbf{3.835} \pm 0.055$	9.996 ± 0.649
IEO-C	22.951 ± 0.567	24.515	0.023 ± 0.006	3.977 ± 0.124	$\mathbf{9.441} \pm 0.252$
ETO-Q	31.485 ± 2.197	38.226	$\mathbf{0.000} \pm 0.001$	$\mathbf{3.835} \pm 0.055$	9.996 ± 0.649
IEO-Q	$\mathbf{22.730} \pm 0.549$	$\mathbf{24.255}$	0.017 ± 0.005	4.012 ± 0.067	12.186 ± 0.831

5.4 Long-term scheduling: Out-of-sample evaluation under strong misspecification

Finally, we examine a long-term scheduling scenario in which all available data pairs are included, rather than limiting our analysis to cases with an RUL of 125 or less. This setting better captures realistic conditions where the decision-maker lacks precise information about equipment health and must therefore proactively schedule maintenance early in the equipment's life. Consequently, maintenance decisions taken later (e.g., after 125 cycles) may be interpreted as proxies for health indicators. To model this scenario, we simulate a two-phase degradation process by manually capping all RUL values greater than 125 at 125, aligning them with the defined latest maintenance window. This adjustment introduces a more complex data distribution, significantly exacerbating model misspecification, as our predictive models are not explicitly designed to capture a two-stage degradation process. As in previous scenarios, we use the first 20 engines for testing and the remaining ones for training, repeating the experiment independently 100 times. The results are illustrated in Figure 5 and summarized in Table 4.

Under stronger model misspecification, all frameworks exhibit reduced performance across all metrics compared to their short-term counterparts. However, the decision quality improvements resulting from the IEO fine-tuning become more prominent. Numerically, the IEO framework reduces average regret by 6.56% when using the CSO policy and by 21.92% with the quantile policy. Consistent with previous findings, the IEO-C and IEO-Q frameworks deliver comparable decision-making performance. Specifically, IEO-C maintains the lowest MAE loss overall, whereas IEO-Q provides the safest decisions in terms of failure frequency, except for the over-conservative and unstable ETO-Q framework. The instability of ETO-Q, reflected in its significantly high variability in regret, stems from unreliable predictive distributions caused by strong model misspecification. In contrast, the IEO approach effectively corrects suboptimal decisions produced by the quantile policy by explicitly incorporating decision-oriented losses into model training, though at the cost of substantially increased NLL and MAE losses.

6 Concluding Remarks

This paper investigates the consistency of machine learning-based predictive maintenance approaches under limited data and model misspecification. We highlight that intuitive ETO approaches inherently lead to suboptimal maintenance decisions due to estimation errors, and propose an IEO framework that maintains consistency in decision-making. We establish the consistency of our IEO method through non-asymptotic probabilistic guarantees. Technically, our proposed framework fine-tunes existing pre-
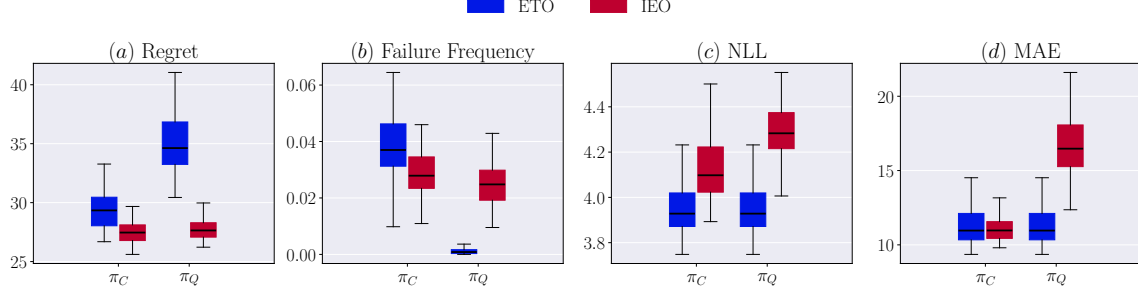
Figure 5: Performance of four frameworks under different evaluation metrics in the long-term scheduling case by 100 experiments. *Note.* The boxes are plotted with the 25% and the 75% confidence intervals. The long-term scheduling case evaluates the metrics over 20 testing engines.

Table 4: Evaluation metrics of four frameworks in the long-term scheduling case. *Note.* Evaluation metrics are demonstrated in mean value $\pm$ the standard deviation, considering 100 independent experiments.

Framework	Regret	Max Regret	Failure Frequency	NLL	MAE
ETO-C	29.446 ± 1.953	36.777	0.037 ± 0.012	$\mathbf{3.947} \pm 0.106$	11.474 ± 1.690
IEO-C	$\mathbf{27.515} \pm 0.989$	31.111	0.028 ± 0.008	4.138 ± 0.157	$\mathbf{11.112} \pm 0.921$
ETO-Q	35.556 ± 3.915	51.113	$\mathbf{0.001} \pm 0.002$	$\mathbf{3.947} \pm 0.106$	11.474 ± 1.690
IEO-Q	27.759 ± 0.935	$\mathbf{30.758}$	0.025 ± 0.008	4.297 ± 0.138	16.745 ± 2.374

dictive models, and we introduce a stochastic perturbation gradient descent algorithm to manage the non-differentiability challenges associated with discrete optimization problems. Extensive experiments conducted on aero-engine datasets demonstrate the robustness and consistency of the IEO framework under varying degrees of model misspecification. These experiments provide empirical evidence that the proposed IEO approach consistently outperforms the ETO framework in terms of decision quality, particularly as model misspecification intensifies, with only modest trade-offs in predictive accuracy. These findings align closely with recent theoretical advances in data-driven optimization (Elmachtoub et al., 2023, 2025).

Despite its advantages, the IEO framework inevitably incurs increased computational complexity compared to ETO methods, primarily because it requires iterative solving of multiple optimization instances. To mitigate this computational burden, we employ a stochastic perturbation algorithm combined with low-dimensional uncertainty representations. Nevertheless, future research could explore alternative end-to-end learning techniques tailored to complex optimization scenarios. Additionally, applying the IEO framework to more complex maintenance problems, such as joint ordering-and-replacing, multi-component maintenance under resource constraints, and decision-dependent maintenance, represents important directions for further studies. Finally, advancing theoretical insights specifically tailored to small-data contexts and dependent data would significantly enhance the practical applicability of the IEO approach, addressing a notable gap in current literature that primarily focuses on asymptotic and independent scenarios.

A Proofs of Theorem and Lemma

Definition 4 (Natarajan Dimension). *Consider $\mathcal{G} : \mathcal{X} \to \mathcal{Z}$ a set of functions that maps contextual information $x \in \mathcal{X}$ to a feasible decision $z \in \mathcal{Z}$. $\mathcal{G}$ shatters a set $\mathcal{X}' \subset \mathcal{X}$ if there exists two functions $g_1, g_2 \in \mathcal{G}$ such that:*

1. $\forall x \in \mathcal{X}', g_1(x) \neq g_2(x)$.

2. $\forall \mathcal{X}'' \subset \mathcal{X}', \exists g \in \mathcal{G}$, such that $\forall x \in \mathcal{X}'', g(x) = g_1(x)$, and $\forall x \in \mathcal{X}' \backslash \mathcal{X}'', g(x) = g_2(x)$.

The Natarajan dimension of $\mathcal{G}$ is the maximal cardinality of a set $\mathcal{X}'$ that can be shattered by $\mathcal{G}$.

Proof. Proof of Theorem 1. Following the notation in Section 3, we denote the unified policy class as $\mathcal{G}$. Define the empirical Rademacher complexity of class $\mathcal{G}$ given dataset $\mathcal{D}$ as

$$\mathcal{R}_{\mathcal{D}}(\mathcal{G}) \triangleq \frac{1}{n}\mathbb{E}_\sigma \left[\sup_{g_\omega \in \mathcal{G}} \sum_{i=1}^n \sigma_i c(g_\omega(x_i), y_i) \right],$$

where σ is the n-dimensional independent Rademacher random variable and c the optimization objective function. Next, denote the growth function of model class $\mathcal{G}$ as

$$\Pi_{\mathcal{G}}(n) \triangleq \max_{(x_1,\dots,x_n)\in\mathcal{X}} |\{(g_\omega(x_1), \dots, g_\omega(x_n)), g_\omega \in \mathcal{G}\}|,$$

which indicates the maximum number of unique n-dimensional decision vectors that can be induced by class $\mathcal{G}$. By the multiclass generalization of Sauer's lemma (Haussler and Long, 1995, Corollary 5), if $n \geq d$ and $K > 2$, then

$$\Pi_{\mathcal{G}}(n) \leq \sum_{i=0}^d \binom{n}{i} \binom{K+1}{2}^i \leq \left(\frac{en}{d} \frac{K(K+1)}{2} \right)^d \leq \left(\frac{enK^2}{d} \right)^d$$

where d denotes the Natarajan dimension of class $\mathcal{G}$ that is irrelevant to data size n.

Further, we denote the set of all decision vectors that can be induced by $\mathcal{G}$ and a fixed dataset $\mathcal{D}$ as

$$\mathcal{Z}_{\mathcal{D}} \triangleq \{(g_\omega(x_1), \dots, g_\omega(x_n)), g_\omega \in \mathcal{G}\},$$

which has an upper bound $|\mathcal{Z}_{\mathcal{D}}| \leq \Pi_{\mathcal{G}}(n)$ for every $\mathcal{D} \sim \mathbb{P}^n$ by definition. Therefore,

$$\begin{aligned}
\mathcal{R}_{\mathcal{D}}(\mathcal{G}) &= \frac{1}{n}\mathbb{E}_\sigma \left[\sup_{g_\omega \in \mathcal{G}} \sum_{i=1}^n \sigma_i c(g_\omega(x_i), y_i) \right] \\
&= \frac{1}{n}\mathbb{E}_\sigma \left[\sup_{(z_1,\dots,z_n)\in\mathcal{Z}_{\mathcal{D}}} \sum_{i=1}^n \sigma_i c(z_i, y_i) \right] \\
&\leq C_1 \sqrt{n} \frac{\sqrt{2\log|\mathcal{Z}_{\mathcal{D}}|}}{n} \\
&= C_1 \sqrt{\frac{2\log|\mathcal{Z}_{\mathcal{D}}|}{n}} \\
&\leq C_1 \sqrt{\frac{4d\log K + 2d\log(en/d)}{n}}
\end{aligned} \tag{A.1}$$

The first inequality is by the Massart's lemma (Mohri et al., 2018, Theorem 3.7), the boundedness of c, and the invariance of y_i given dataset $\mathcal{D}$. The second inequality is by the Sauer's lemma.

An alternative exists to drop the requirement for $d \leq n$. Instead of using the Sauer's lemma to bound $\Pi_{\mathcal{G}}(n)$, one can consider the classic Natarajan lemma (Shalev-Shwartz and Ben-David, 2014, lemma 29.4), which states that

$$\Pi_{\mathcal{G}}(n) \leq n^d K^{2d},$$

therefore

$$C_1 \sqrt{\frac{2\log|\mathcal{Z}_{\mathcal{D}}|}{n}} \leq C_1 \sqrt{\frac{4d\log K + 2d\log n}{n}}$$

is valid no matter the relationship between d and n. This bound is adopted by El Balghiti et al. (2023).

Still consider the case where $d \leq n$. Now the empirical Rademacher complexity $\mathcal{R}_{\mathcal{D}}(\mathcal{G})$ has an data-irrelevant upper bound by (A.1), we can use the same upper bound again for the Rademacher complexity of $\mathcal{G}$:

$$\mathcal{R}_n(\mathcal{G}) \triangleq \mathbb{E}_{\mathcal{D} \sim \mathbb{P}^n}[\mathcal{R}_{\mathcal{D}}(\mathcal{G})] \leq C_1 \sqrt{\frac{4d \log K + 2d \log(en/d)}{n}} \tag{A.2}$$

For i.i.d. sample $\mathcal{D}$ drawn from $\mathbb{P}^n$, by Mohri et al. (2018, Theorem 3.3), with probability at least $1 - \delta$,

$$R_L(\omega) \leq \widehat{R}_L(\omega) + 2\mathcal{R}_n(\mathcal{G}) + C_1 \sqrt{\frac{\log(1/\delta)}{2n}}. \tag{A.3}$$

Replacing the empirical Rademacher complexity in (A.3) by the upper bound in (A.2) finishes the proof. $\qquad\square$

Proof. Proof of Corollary 1. Denote

$$\omega_{erm} \in \arg\min_{\omega \in \Omega} \widehat{R}_L(\omega),$$
$$\omega^\star \in \arg\min_{\omega \in \Omega} R_L(\omega).$$

By definition $\widehat{R}_L(\omega_{erm}) \leq \widehat{R}_L(\omega^\star)$, thus the following inequality holds with probability 1:

$$\begin{aligned}
R_L(\omega_{erm}) - R_L(\omega^\star) &= R_L(\omega_{erm}) - \widehat{R}_L(\omega_{erm}) \\
&\quad + \widehat{R}_L(\omega^\star) - R_L(\omega^\star) \\
&\quad + \widehat{R}_L(\omega_{erm}) - \widehat{R}_L(\omega^\star) \\
&\leq R_L(\omega_{erm}) - \widehat{R}_L(\omega_{erm}) \\
&\quad + \widehat{R}_L(\omega^\star) - R_L(\omega^\star)
\end{aligned} \tag{A.4}$$

By Assumptions 1-2, applying Hoeffding's inequality to $\widehat{R}_L(\omega^\star) - R_L(\omega^\star)$ gives:

$$\mathbb{P}^n\{\widehat{R}_L(\omega^\star) - R_L(\omega^\star) \geq \epsilon\} \leq \exp\left\{-\frac{n\epsilon^2}{2C_1^2}\right\},$$

Taking $\delta/2 = \exp\left\{-\frac{n\epsilon^2}{2C_1^2}\right\}$ further leads to:

$$\mathbb{P}^n\left\{\widehat{R}_L(\omega^\star) - R_L(\omega^\star) \leq C_1\sqrt{\frac{\log(2/\delta)}{2n}}\right\} \geq 1 - \delta/2, \tag{A.5}$$

Applying Theorem 1 to ω_{erm} guarantees that with probability at least $1 - \delta/2$,

$$\begin{aligned}
R_L(\omega_{erm}) &\leq \widehat{R}_L(\omega_{erm}) + C_1\sqrt{\frac{\log(2/\delta)}{2n}} \\
&\quad + C_1\sqrt{\frac{4d \log K + 2d \log(en/d)}{n}}.
\end{aligned} \tag{A.6}$$

Combining the inequalities (A.4)–(A.6) finishes the proof. $\qquad\square$

References

Atamuradov, Vepa, Kamal Medjaher, Pierre Dersin, Benjamin Lamoureux, Noureddine Zerhouni. 2017. Prognostics and health management for maintenance practitioners-review, implementation and tools evaluation. *International Journal of Prognostics and Health Management* **8**(3) 1–31.

Ban, Gah-Yi, Cynthia Rudin. 2019. The big data newsvendor: Practical insights from machine learning. *Operations Research* **67**(1) 90–108.

Bartlett, Peter L, Andrea Montanari, Alexander Rakhlin. 2021. Deep learning: a statistical viewpoint. *Acta numerica* **30** 87–201.

Basciftci, Beste, Shabbir Ahmed, Nagi Gebraeel. 2020. Data-driven maintenance and operations scheduling in power systems under decision-dependent uncertainty. *IISE transactions* **52**(6) 589–602.

Berthet, Quentin, Mathieu Blondel, Olivier Teboul, Marco Cuturi, Jean-Philippe Vert, Francis Bach. 2020. Learning with differentiable pertubed optimizers. *Advances in Neural Information Processing Systems* **33** 9508–9519.

Bertsimas, Dimitris, Nathan Kallus. 2020. From predictive to prescriptive analytics. *Management Science* **66**(3) 1025–1044.

Cai, Yue, Ruud H Teunter, Bram de Jonge. 2023. A data-driven approach for condition-based maintenance optimization. *European Journal of Operational Research* **311**(2) 730–738.

Canavos, George C, Chris P Taokas. 1973. Bayesian estimation of life parameters in the weibull distribution. *Operations Research* **21**(3) 755–763.

De Jonge, Bram, Philip A Scarf. 2020. A review on maintenance optimization. *European Journal of Operational Research* **285**(3) 805–824.

Drent, Collin, Melvin Drent, Joachim Arts. 2024. Condition-based production for stochastically deteriorating systems: optimal policies and learning. *Manufacturing & Service Operations Management* **26**(3) 1137–1156.

Drent, Collin, Melvin Drent, Joachim Arts, Stella Kapodistria. 2023. Real-time integrated learning and decision making for cumulative shock degradation. *Manufacturing & Service Operations Management* **25**(1) 235–253.

El Balghiti, Othman, Adam N Elmachtoub, Paul Grigas, Ambuj Tewari. 2023. Generalization bounds in the predict-then-optimize framework. *Math. Oper. Res.* **48**(4) 2043–2065.

Elmachtoub, Adam N, Paul Grigas. 2022. Smart "predict, then optimize". *Management Science* **68**(1) 9–26.

Elmachtoub, Adam N., Henry Lam, Haixiang Lan, Haofeng Zhang. 2025. Dissecting the impact of model misspecification in data-driven optimization. *The 28th International Conference on Artificial Intelligence and Statistics*.

Elmachtoub, Adam N, Henry Lam, Haofeng Zhang, Yunfan Zhao. 2023. Estimate-then-optimize versus integrated-estimation-optimization versus sample average approximation: a stochastic dominance perspective. *arXiv preprint arXiv:2304.06833* .

Gebraeel, Nagi, Yaguo Lei, Naipeng Li, Xiaosheng Si, Enrico Zio, et al. 2023. Prognostics and remaining useful life prediction of machinery: advances, opportunities and challenges. *Journal of Dynamics, Monitoring and Diagnostics* 1–12.

Haussler, David, Philip M Long. 1995. A generalization of sauer's lemma. *Journal of Combinatorial Theory, Series A* **71**(2) 219–240.

Ho-Nguyen, Nam, Fatma Kılınç-Karzan. 2022. Risk guarantees for end-to-end prediction and optimization processes. *Management Science* **68**(12) 8680–8698.

Hu, Yichun, Nathan Kallus, Xiaojie Mao. 2022. Fast rates for contextual linear optimization. *Management Science* **68**(6) 4236–4245.

Kamariotis, Antonios, Konstantinos Tatsis, Eleni Chatzi, Kai Goebel, Daniel Straub. 2024. A metric for assessing and optimizing data-driven prognostic algorithms for predictive maintenance. *Reliability Engineering & System Safety* **242** 109723.

Kannan, Rohit, Güzin Bayraksan, James R Luedtke. 2025. Data-driven sample average approximation with covariate information. *Operations Research* .

Kingma, Diederik P, Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* .

Lagos, Carlos, Felipe Delgado, Mathias A Klapp. 2020. Dynamic optimization for airline maintenance operations. *Transportation Science* **54**(4) 998–1015.

Lee, Hyunju, Ji Hwan Cha. 2016. New stochastic models for preventive maintenance and maintenance optimization. *European Journal of Operational Research* **255**(1) 80–90.

Lee, Juseong, Mihaela Mitici. 2023. Deep reinforcement learning for predictive aircraft maintenance using probabilistic remaining-useful-life prognostics. *Reliability Engineering & System Safety* **230** 108908.

Lei, Yaguo, Naipeng Li, Liang Guo, Ningbo Li, Tao Yan, Jing Lin. 2018. Machinery health prognostics: A systematic review from data acquisition to rul prediction. *Mechanical systems and signal processing* **104** 799–834.

Lewis, Austin D, Katrina M Groth. 2022. Metrics for evaluating the performance of complex engineering system health monitoring models. *Reliability Engineering & System Safety* **223** 108473.

Li, Zhe, Qian He, Jingyue Li. 2024. A survey of deep learning-driven architecture for predictive maintenance. *Engineering applications of artificial intelligence* **133** 108285.

Lin, Shaochong, Youhua Chen, Yanzhi Li, Zuo-Jun Max Shen. 2022. Data-driven newsvendor problems regularized by a profit risk constraint. *Production and Operations Management* **31**(4) 1630–1644.

Lv, Jinchi, Jun S Liu. 2014. Model selection principles in misspecified models. *Journal of the Royal Statistical Society Series B: Statistical Methodology* **76**(1) 141–167.

Mandi, Jayanta, James Kotary, Senne Berden, Maxime Mulamba, Victor Bucarey, Tias Guns, Ferdinando Fioretto. 2024. Decision-focused learning: Foundations, state of the art, benchmark and future opportunities. *Journal of Artificial Intelligence Research* **80** 1623–1701.

Mitici, Mihaela, Ingeborg de Pater, Anne Barros, Zhiguo Zeng. 2023. Dynamic predictive maintenance for multiple components using data-driven probabilistic rul prognostics: The case of turbofan engines. *Reliability Engineering & System Safety* **234** 109199.

Mohamed, Shakir, Mihaela Rosca, Michael Figurnov, Andriy Mnih. 2020. Monte carlo gradient estimation in machine learning. *Journal of Machine Learning Research* **21**(132) 1–62.

Mohri, Mehryar, Afshin Rostamizadeh, Ameet Talwalkar. 2018. *Foundations of Machine Learning*. 2nd ed. MIT Press, Cambridge, MA.

Nguyen, Khanh TP, Kamal Medjaher. 2019. A new dynamic predictive maintenance framework using deep learning for failure prognostics. *Reliability Engineering & System Safety* **188** 251–262.

Papamakarios, George, Eric Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, Balaji Lakshminarayanan. 2021. Normalizing flows for probabilistic modeling and inference. *Journal of Machine Learning Research* **22**(57) 1–64.

Paszke, Adam, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 32. Curran Associates, Inc.

Pintelon, Liliane M, LF Gelders. 1992. Maintenance management decision making. *European Journal of Operational Research* **58**(3) 301–317.

Qi, Meng, Paul Grigas, Zuo-Jun Max Shen. 2021. Integrated conditional estimation-optimization. *arXiv preprint arXiv:2110.12351* .

Qi, Meng, Yuanyuan Shi, Yongzhi Qi, Chenxin Ma, Rong Yuan, Di Wu, Zuo-Jun Shen. 2023. A practical end-to-end inventory management model with deep learning. *Management Science* **69**(2) 759–773.

Sadana, Utsav, Abhilash Chenreddy, Erick Delage, Alexandre Forel, Emma Frejinger, Thibaut Vidal. 2025. A survey of contextual optimization methods for decision-making under uncertainty. *European Journal of Operational Research* **320**(2) 271–289.

Saxena, Abhinav, Kai Goebel, Don Simon, Neil Eklund. 2008. Damage propagation modeling for aircraft engine run-to-failure simulation. *2008 international conference on prognostics and health management*. IEEE, 1–9.

Shalev-Shwartz, Shai, Shai Ben-David. 2014. *Understanding machine learning: From theory to algorithms*. Cambridge university press.

Sim, Melvyn, Qinshen Tang, Minglong Zhou, Taozeng Zhu. 2024. The analytics of robust satisficing: predict, optimize, satisfice, then fortify. *Operations Research* .

Srivastava, Nitish, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **15**(1) 1929–1958.

Stratigakos, Akylas, Salvador Pineda, Juan Miguel Morales. 2024. Decision-focused linear pooling for probabilistic forecast combination. *International Journal of Forecasting* .

van der Haar, Joost F, Collin Drent, Robert N Boute. 2025. Predict-and-optimize machine learning for feature-based maintenance optimization. *Available at SSRN 5225643* .

Wagenaar, Joris C, Leo G Kroon, Marie Schmidt. 2017. Maintenance appointments in railway rolling stock rescheduling. *Transportation Science* **51**(4) 1138–1160.

Waldhauser, Felix, Hamza Boukabache, Daniel Perrin, Martin Dazer. 2024. Generating realistic failure data for predictive maintenance: A simulation and cgan-based methodology. *PHM Society European Conference* **8**(1) 4.

Wen, Yuxin, Md Fashiar Rahman, Honglun Xu, Tzu-Liang Bill Tseng. 2022. Recent advances and trends of predictive maintenance from data-driven machine prognostics perspective. *Measurement* **187** 110276.

Williams, Ronald J. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning* **8** 229–256.

Zhuang, Liangliang, Ancha Xu, Xiao-Lin Wang. 2023. A prognostic driven predictive maintenance framework based on bayesian deep learning. *Reliability Engineering & System Safety* **234** 109181.