DECISION-FOCUSED CLUSTERING

Anonymous authors
Paper under double-blind review

ABSTRACT

Modern platforms often need to make personalized decisions for a large number of instances, repeatedly solving the same optimization problem with different instance-specific parameters. When the number of distinct decisions is limited, a natural approach is to group instances into clusters and assign a shared decision to all instances within each cluster. Standard clustering methods such as k-means, however, group instances according to geometric similarity, which need not preserve downstream decision quality. We propose *Decision-Focused Clustering* (DFC) for linear downstream optimization problems, which forms clusters based directly on the quality of the decisions shared within each cluster. To formalize this objective, we introduce a decision loss that measures the regret of using the decision optimized for one cost vector when the true cost vector is another. Since the resulting clustering problem is NP-hard, we develop an alternating optimization algorithm that iteratively updates cluster representatives and instance assignments. We further establish theoretical convergence guarantees for the proposed algorithm. We evaluate our framework on synthetic shortest-path and portfolio optimization problems and a real-world courier dispatching problem, covering decision problems with different feasible-set structures, including polyhedral and strongly convex settings. Compared with k-means, DFC consistently achieves lower decision loss in all three application settings.

1 INTRODUCTION

Clustering is an unsupervised learning technique for summarizing large datasets through a small set of representative groups. When instances within each cluster share a common decision, clustering can reduce the number of downstream optimization problems that must be solved. Methods such as k-means form clusters by minimizing the within-cluster squared Euclidean distance. However, when clusters are later used to support optimization decisions, geometric similarity does not necessarily indicate whether instances can share a good decision. Two nearby cost vectors can lead to different optimal decisions, whereas distant vectors can lead to the same one.

In this paper, we study how clustering should be designed when instances within each cluster share a downstream optimization decision. We propose Decision-Focused Clustering (DFC), which forms clusters according to the quality of the decisions shared within them. Specifically, instead of assigning an instance to its nearest centroid, we assign it to the representative that yields the smallest decision loss. Decision loss measures the regret from using the decision optimized for one cost vector when performance is evaluated under another.

Why not cluster decisions? A natural alternative is to simply choose the K most frequently occurring optimal decisions and use them as the cluster representatives. However, identifying which decisions occur most frequently requires first computing the individualized optimal decision for every instance. DFC does not require this preprocessing step. Although decision loss is defined relative to each instance's individualized optimum, this term is constant across candidate clusters for a given instance, so cluster assignment only depends on the decisions induced by the cluster representatives. More importantly, this approach relies on optimal decisions repeating across instances, which need not hold in general. For polyhedral feasible regions, the number of candidate extreme-point solutions can be very large, and for more general convex feasible regions, the optimal decision may vary continuously with the cost vector. Thus, clustering based on repeated decisions or decision frequencies does not provide a general solution.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

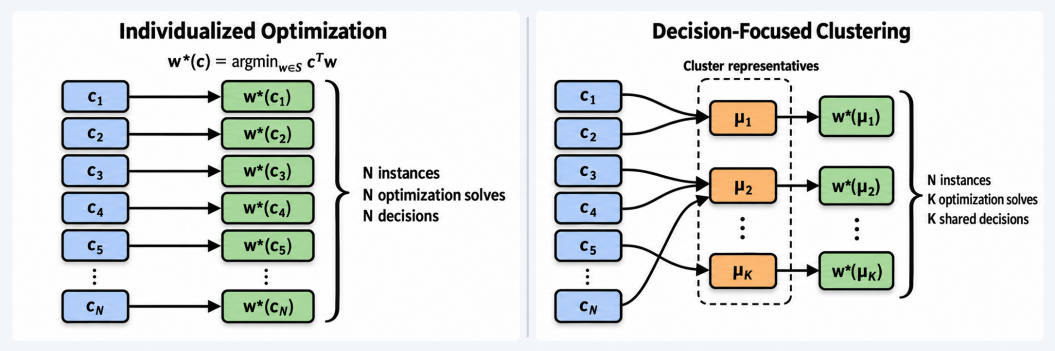


Figure 1: Overview of DFC compared with individualized optimization. Individualized optimization solves a separate problem for each instance, whereas DFC represents N instances using $K \ll N$ representative cost vectors whose induced optimal decisions are shared within clusters.

Geometrically clustering the individualized decisions does not resolve this issue either, since proximity in the decision space does not generally correspond to the loss incurred from sharing a decision. We therefore formulate the problem through representative cost vectors. For each cluster, we seek a representative whose corresponding optimal decision minimizes the aggregate decision loss over the assigned instances. We show that the decision optimized for the cluster mean minimizes the within-cluster decision loss. The main contributions of this paper are as follows:

- We introduce Decision-Focused Clustering (DFC) for clustering instances that share downstream optimization decisions.
- We formulate the clustering objective as the aggregate decision loss so that cluster assignments are determined by the quality of the shared decisions, which covers decision problems with different feasible-set structures, including polyhedral and strongly convex settings.
- We show that the resulting clustering problem is NP-hard and we develop an alternating optimization algorithm with closed-form representative updates whose objective values decrease monotonically and converge. We further establish statistical guarantees for the empirical DFC problem under the feasible-set settings considered in the paper.
- We evaluate DFC on synthetic shortest-path and portfolio optimization problems and a real-world Meituan courier-dispatching problem. Across all three settings, DFC achieves lower held-out decision loss than k-means.

Literature review. Decision-focused learning estimates unknown parameters in stochastic optimization problems by aligning predictive models with downstream decision quality rather than prediction accuracy. Existing approaches support efficient training through differentiable optimization and related optimization-based methods (Donti et al., 2017; Amos & Kolter, 2017; Agrawal et al., 2019; Wilder et al., 2019; Ferber et al., 2020; Mandi et al., 2020; Mandi & Guns, 2020; Hu et al., 2023), as well as decision-aware surrogate losses (Elmachtoub & Grigas, 2022; Gupta & Huang, 2024; Wan & Liu, 2026; Lee et al., 2026). See Mandi et al. (2024); Sadana et al. (2025) for comprehensive reviews. Beyond model training, decision-focused ideas guide data collection (Liu et al., 2023; Wan et al., 2025) and representative-parameter design (Er & Liu, 2025).

Classical k-means minimizes within-cluster squared Euclidean distance (Lloyd, 1982). Banerjee et al. (2005) generalize this approach to Bregman divergences generated by strictly convex, differentiable functions, retaining mean-based representative updates and monotone alternating optimization. DFC shares this algorithmic structure but derives its dissimilarity from downstream decision regret. Specifically, our loss admits a generalized Bregman representation generated by the negative optimal-value function, which is not strictly convex and may be nonsmooth. Consequently, distinct cost vectors can incur zero loss when they induce the same optimal decision, allowing clustering to preserve decision quality without preserving geometric proximity.

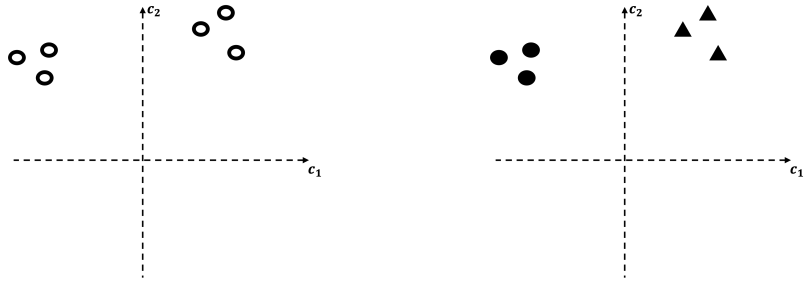
Relatedly, Li et al. (2023) exploit cluster structure to improve data aggregation across optimization problems with limited data. Our setting instead uses clustering to reduce individualized optimization to a limited number of shared decisions, grouping cost vectors according to the decision loss incurred by sharing those decisions.

2 Problem Setting

We consider N instances represented by cost vectors $c_1, \ldots, c_N \in \mathbb{R}^d$, where each c_i serves as the objective coefficient vector of a linear optimization problem over a common feasible region $S \subseteq \mathbb{R}^d$. We allow S to be either a polyhedron or a strongly convex set, encompassing a broad class of decision-making problems, including *shortest-path, inventory management, portfolio optimization, label ranking, and multiclass classification* (Sadana et al., 2025). For a cost vector $c \in \mathbb{R}^d$, a feasible decision is denoted by $w \in S$, with objective cost $c^\top w$, and we denote the selected optimal decision by $w^*(c) = \arg\min_{w \in S} c^\top w$. When multiple optimal solutions exist, $w^*(c)$ denotes a selected optimum under a fixed selection rule that makes the mapping single-valued. We use $D := \mathrm{diam}(S) > 0$ to denote the diameter of the feasible region. Our goal is to partition the N instances into K clusters, indexed by $j \in [K]$, such that all instances assigned to the same cluster share a common decision.

2.1 Decision Loss

Traditional clustering methods partition the N instances into K clusters based on geometric similarity in the parameter space. In many decision-making settings, however, instances that are geometrically similar need not induce similar decisions, while instances that are far apart may still admit nearly identical decisions. As an illustrative two-dimensional example, consider Figure 2, where six points are partitioned into two clusters based on geometric proximity.



(a) Original data distribution. (b) Two clusters obtained by a traditional clustering method such as k-means.

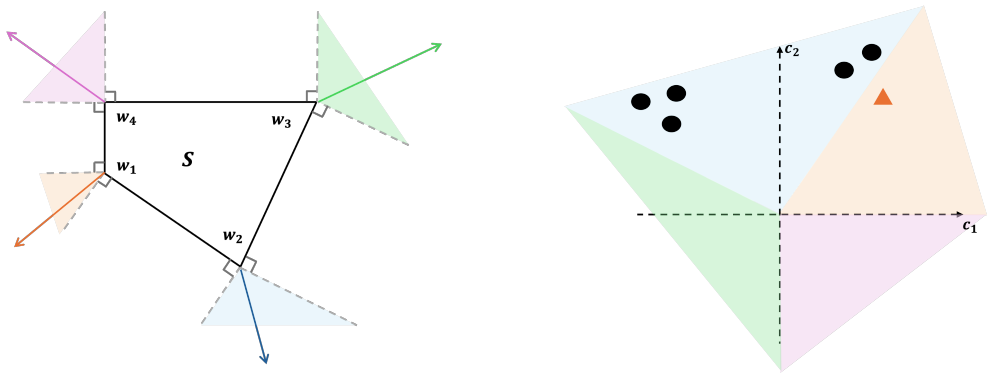
Figure 2: Illustration of traditional clustering based on geometric similarity.

Suppose now that these six data points represent two-dimensional coefficient vectors for a downstream linear programming problem, whose feasible region is shown in Figure 3(a). Compared with Figure 2(b), the clustering result in Figure 3(b) is better aligned with the downstream decision-making objective. To generalize this idea to linear optimization problems with more general feasible regions, such as strongly convex sets, we use the *Smart Predict-then-Optimize* (SPO) regret (Elmachtoub & Grigas, 2022) as our measure of decision loss throughout the paper:

$$\ell_{\mathrm{decision}}(\hat{c}, c) := c^\top \big(w^*(\hat{c}) - w^*(c)\big).$$

The first argument, $\hat{c}$, determines the decision $w^*(\hat{c})$, while the second argument, c, evaluates that decision under the true cost vector. Thus, ℓ_{decision} is asymmetric. In particular, if $w^*(\hat{c}) = w^*(c)$, then $\ell_{\mathrm{decision}}(\hat{c}, c) = 0$, regardless of the geometric distance between $\hat{c}$ and c.

Clustering directly by the decision regions illustrated in Figure 3 has two limitations. First, a strongly convex feasible region can have infinitely many extreme points, so the coefficient space need not admit a finite partition into regions sharing the same optimal decision. Second, even for a polyhedral feasible region, the prescribed number of clusters K may be much smaller than the number of



(a) A linear program partitions the coefficient space into decision regions, each associated with an extreme point of the feasible set.

(b) Clustering induced by the decision regions. The five circles share the optimal decision w_2, whereas the triangle has optimal decision w_1.

Figure 3: Illustration of DFC for a downstream linear programming problem.

extreme points. This raises the question of how to aggregate decision regions and select a shared decision for each resulting cluster while minimizing the induced decision loss.

Let $\mu_j \in \mathbb{R}^d$ denote the representative cost vector of cluster j, for $j = 1, \ldots, K$. In DFC, the representative μ_j appears in the first argument of the decision loss because it determines the shared decision $w^*(\mu_j)$ used by all instances assigned to that cluster. Thus, assigning instance i to cluster j incurs decision loss:

$$\ell_{\text{decision}}(\mu_j, c_i) = c_i^\top \big(w^*(\mu_j) - w^*(c_i)\big). \tag{1}$$

2.2 Standard k-means Baseline

Standard k-means alternates between assigning instances to their nearest centroids and updating cluster representatives. It learns K centroids $\{\mu_j\}_{j=1}^K$ by minimizing within-cluster Euclidean distance: $\min_{\mu_1,\ldots,\mu_K} \sum_{i=1}^N \min_{j=1,\ldots,K} \|c_i - \mu_j\|^2$. The next section adapts this assignment-and-update structure to a clustering objective defined by decision loss.

3 Decision-Focused Clustering Framework

For each instance i and cluster j, let $\gamma_{ij} \in \{0, 1\}$ indicate whether instance i is assigned to cluster j. Each instance is assigned to exactly one cluster, so $\sum_{j=1}^K \gamma_{ij} = 1$ for all $i = 1, \ldots, N$. Let $\mathbf{\Gamma} = [\gamma_{ij}]$ denote the assignment matrix. The DFC objective is

$$\min_{\mu_1,\ldots,\mu_K, \mathbf{\Gamma}} \sum_{i=1}^N \sum_{j=1}^K \gamma_{ij} \ell_{\text{decision}}(\mu_j, c_i). \tag{2}$$

Equation 2 measures the total decision loss incurred when each instance uses the shared decision of its assigned cluster. For notational convenience, let $\boldsymbol{\mu} = [\mu_1, \ldots, \mu_K]^\top$ and define $\mathcal{L}_{\text{DFC}}(\mathbf{\Gamma}, \boldsymbol{\mu}) = \sum_{i=1}^N \sum_{j=1}^K \gamma_{ij} \ell_{\text{decision}}(\mu_j, c_i)$.

Theorem 1 shows that the DFC problem is NP-hard in general; see Appendix A for the proof.

Theorem 1. *Problem (2) is NP-hard in general, even when $K = 2$, the downstream feasible region is $S = [0, 1]^d$, and $c_i \in \{-1, 0, 1\}^d$ for every instance i.*

Because of the NP-hardness established in Theorem 1, we focus on developing a computationally efficient heuristic algorithm for DFC. A key observation is that, for a fixed assignment matrix $\mathbf{\Gamma}$, the objective in equation 2 decomposes across clusters $j \in [K]$, allowing each cluster representative to be updated independently. Specifically, updating the representative μ_j amounts to minimizing the aggregate decision loss over the instances assigned to cluster j. Unlike in k-means, where the

optimal cluster representative is the centroid of the assigned data points, it is not immediately clear whether the same property holds under decision loss. Proposition 1 shows that the mean of the cost vectors within each cluster is indeed an optimal representative for minimizing the aggregate decision loss.

Proposition 1 (Optimal Representative Update)**.** *For a fixed nonempty cluster $\mathcal{C}_j = \{i : \gamma_{ij} = 1\}$, define the cluster mean $\bar{c}_j = \frac{1}{|\mathcal{C}_j|}\sum_{i\in\mathcal{C}_j} c_i$. Then $\bar{c}_j \in \arg\min_{\mu\in\mathbb{R}^d}\sum_{i\in\mathcal{C}_j}\ell_{\text{decision}}(\mu, c_i)$. In particular, $w^*(\bar{c}_j)$ is an optimal shared decision for the fixed cluster.*

Proposition 1 provides a closed-form representative update for each nonempty cluster. For fixed representatives, minimizing the DFC objective over the assignments assigns each instance to the representative with the smallest decision loss. For fixed assignments, Proposition 1 gives the optimal representative update as the mean of the assigned cost vectors. Algorithm 1 alternates between these two updates. Since $c_i^\top w^*(c_i)$ is constant across candidate clusters for a fixed instance i, minimizing decision loss over j is equivalent to minimizing $c_i^\top w^*(\mu_j)$. Thus, the assignment step does not require computing the individualized decision $w^*(c_i)$.

Algorithm 1 Decision-Focused Clustering

Input: cost vectors $c_1, \ldots, c_N$, number of clusters K, maximum number of iterations $T_{\max}$, and tolerance τ.
Initialization: initialize $\boldsymbol{\mu}^0 = [\mu_1^0, \ldots, \mu_K^0]^\top$ and assignments $\boldsymbol{\Gamma}^0$ using k-means.
for $t = 0, 1, \ldots, T_{\max} - 1$ **do**
  for $j = 1, \ldots, K$ **do**
    $w_j^t \leftarrow w^*(\mu_j^t)$. ▷ *Compute the decision for cluster j*
  end for
  for $i = 1, \ldots, N$ **do**
    $j_i^{t+1} \in \arg\min_{j=1,\ldots,K} c_i^\top w_j^t$.
    $\gamma_{ij}^{t+1} \leftarrow \mathbb{I}\{j = j_i^{t+1}\}$, for $j = 1, \ldots, K$. ▷ *Update cluster assignment for instance i*
  end for
  for $j = 1, \ldots, K$ **do**
    $\mathcal{C}_j^{t+1} \leftarrow \{i : \gamma_{ij}^{t+1} = 1\}$.
    if $\mathcal{C}_j^{t+1} = \varnothing$ **then**
      reinitialize μ_j^{t+1} using a randomly selected cost vector.
    else
      $\mu_j^{t+1} \leftarrow \dfrac{1}{|\mathcal{C}_j^{t+1}|}\displaystyle\sum_{i\in\mathcal{C}_j^{t+1}} c_i$. ▷ *Update cluster representatives*
    end if
  end for
  if $\|\boldsymbol{\mu}^{t+1} - \boldsymbol{\mu}^t\|_2 < \tau$ **then**
    break.
  end if
end for
Output: final representatives $\boldsymbol{\mu}$ and assignments $\boldsymbol{\Gamma}$.

Theorem 2 below guarantees convergence of objective values for Algorithm 1. With k-means initialization, monotonicity in Theorem 2 ensures that the empirical decision loss cannot exceed that of the initial k-means partition evaluated using its shared decisions.

Theorem 2 (Convergence of the DFC Objective Values)**.** *At every iteration of Algorithm 1, $\mathcal{L}_{\text{DFC}}(\boldsymbol{\Gamma}^{t+1}, \boldsymbol{\mu}^{t+1}) \le \mathcal{L}_{\text{DFC}}(\boldsymbol{\Gamma}^t, \boldsymbol{\mu}^t)$. Consequently, the sequence of DFC objective values is non-increasing and converges to a finite limit.*

Theorem 2 provides a monotonicity guarantee analogous to that established for Bregman clustering (Banerjee et al., 2005). In practice, multiple initializations can reduce sensitivity to the starting point and improve solution quality, although they do not guarantee global optimality. To separate statistical estimation error from computational optimization error, the analysis in Section 4 studies the statistical performance of DFC under the assumption of exact empirical minimization.

4 STATISTICAL GUARANTEES

Theorem 2 establishes convergence of the empirical objective values generated by our algorithm. We now study the statistical performance of DFC under exact minimization of equation 2. Suppose that $c_1, c_2, \ldots$ are i.i.d. samples from a distribution P satisfying $\mathbb{E}_{c\sim P}\|c\|_2 < \infty$, with d and K fixed. Given representative cost vectors $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_K) \in (\mathbb{R}^d)^K$, we assign a new instance c to the cluster whose shared decision incurs the lowest cost under c. The resulting decision loss is

$$\mathcal{L}(\boldsymbol{\mu}, c) := \min_{j\in[K]} \ell_{\text{decision}}(\mu_j, c) = \min_{j\in[K]} c^\top \big(w^*(\mu_j) - w^*(c)\big).$$

The corresponding population risk is $R(\boldsymbol{\mu}) := \mathbb{E}_{c\sim P}[\mathcal{L}(\boldsymbol{\mu}, c)]$. Let $\widehat{\boldsymbol{\mu}}_N$ be any exact minimizer of the empirical risk $N^{-1}\sum_{i=1}^N \mathcal{L}(\boldsymbol{\mu}, c_i)$, and let $R^* := \min_{\boldsymbol{\mu}\in(\mathbb{R}^d)^K} R(\boldsymbol{\mu})$ denote the optimal population risk with K clusters. We study the excess decision risk $R(\widehat{\boldsymbol{\mu}}_N) - R^*$ as a function of N. Write $D := \operatorname{diam}(S)$, which is finite under either theorem below.

Theorem 3 (Finite identification for polyhedral feasible regions)**.** *Suppose that S is a nonempty bounded polyhedron with vertex set $V = \{v_1, \ldots, v_M\}$, and that $w^*(c)$ always selects an optimal vertex. Then, almost surely, there exists a finite random N_0 such that*

$$R(\widehat{\boldsymbol{\mu}}_N) - R^* = 0 \qquad \textit{for all } N \geq N_0. \tag{3}$$

For fixed K, Theorem 3 shows that exact empirical minimization eventually attains the optimal population risk, almost surely. This is zero excess risk, not necessarily zero absolute decision loss. Finite identification follows from the finite number of vertex tuples and the strictly positive minimum gap between optimal and suboptimal population risks. The random threshold N_0 depends on K, P and the geometry of S.

Assumption 1 (Strongly convex level set)**.** *Let $g : \mathbb{R}^d \to \mathbb{R}$ be σ-strongly convex and L-smooth with respect to the Euclidean norm, where $L \geq \sigma > 0$. Assume that*

$$S = \{w \in \mathbb{R}^d : g(w) \leq r\}, \qquad r > g_{\min} := \min_{w\in\mathbb{R}^d} g(w).$$

This assumption includes Euclidean balls and ellipsoids and ensures that the optimal decision is unique and varies continuously with the cost vector whenever $c \neq 0$; see Liu & Grigas (2021).

Theorem 4 (Consistency for strongly convex feasible regions)**.** *Suppose that S satisfies Assumption 1 and that $\|c\|_2 \leq B$ almost surely under P. Then, for every $N \geq 1$ and $\delta \in (0, 1)$, with probability at least $1 - \delta$,*

$$R(\widehat{\boldsymbol{\mu}}_N) - R^* \leq BD\sqrt{\frac{2\{Kd\log(1+2N) + \log(2/\delta)\}}{N}} + \frac{4BD}{N}.$$

For fixed d and K, Theorem 4 gives a $\widetilde{\mathcal{O}}(N^{-1/2})$ high-probability bound on the excess decision risk. The proof uses only compactness, so the same bound holds for any nonempty compact convex feasible region. Unlike the polyhedral case, there need not be a positive gap between optimal and suboptimal population risks, so finite identification is not guaranteed.

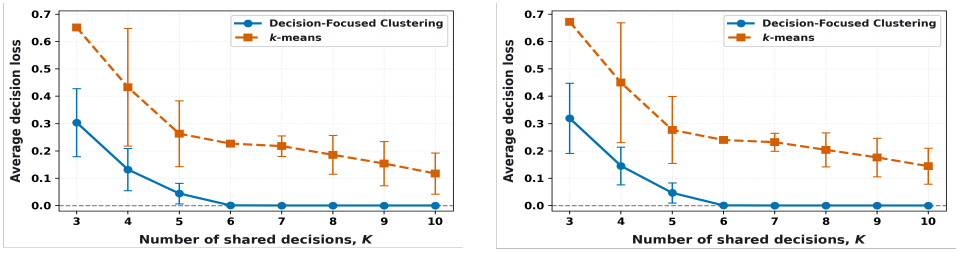
5 EXPERIMENTS

In this section, we evaluate DFC in three complementary settings. The shortest-path experiment examines whether DFC can exploit shared decision structure in a discrete polyhedral problem. The portfolio optimization experiment considers a continuous decision space and studies the effects of compression and training sample size. Finally, the Meituan courier-dispatching experiment evaluates DFC on real-world courier data and held-out instances.

5.1 SHORTEST-PATH EXPERIMENT

We first consider a synthetic shortest-path problem on a directed 3×3 grid. Each instance is represented by a 12-dimensional edge-cost vector $c \in \mathbb{R}^{12}$, and the downstream decision is a source-to-sink path. Under DFC, each representative μ_j induces a shared path $w^*(\mu_j)$ for the instances assigned to cluster j. The full shortest-path formulation is provided in Appendix D.

We generate 4,000 synthetic cost vectors from the same underlying distribution and split them into 3,200 training instances and 800 testing instances. The training set is used to learn the cluster representatives, while the testing set is held out to evaluate whether the learned shared decisions generalize to unseen instances from the same distribution. For each $K \in \{3, \ldots, 10\}$, we run 35 trials and report average decision loss on both sets, computed as $\frac{1}{N}\sum_{i=1}^{N} \min_{j\in[K]} \ell_{\text{decision}}(\mu_j, c_i)$. In each trial, both methods use the same k-means-based initialization obtained from the training set.



(a) Training set.

(b) Testing set.

Figure 4: Average decision loss of DFC and k-means in the shortest-path experiment. Error bars show ± 1 standard deviation over 35 trials.

Because DFC is initialized from the corresponding k-means solution and monotonically decreases the DFC objective, the training-set comparison primarily verifies the optimization behavior (Figure 4a). In the 3×3 grid, each source-to-sink path consists of two right and two downward moves, giving only $\binom{4}{2} = 6$ feasible path decisions. Although instances may differ substantially in their 12-dimensional cost vectors, they fall into at most six groups according to their optimal path decisions. Accordingly, as the number of clusters increases, DFC achieves near-zero decision loss at $K = 6$ and zero loss for $K = 7, \ldots, 10$. In contrast, the decision loss of k-means remains positive because its geometric clusters need not align with the six path decisions.

The testing results in Figure 4b follow the same pattern, indicating that the learned shared decisions remain effective on unseen instances from the same distribution. Averaged over $K = 3, \ldots, 10$, DFC reduces decision loss relative to k-means by 88.3% on training and 87.9% on testing. Restricting to $K = 3, \ldots, 6$, before the saturated regime, the reductions remain 76.5% and 75.9%, respectively. Detailed losses and relative reduction for each value of K are reported in Appendix D.1.

5.2 Portfolio Optimization Experiment

We next consider a portfolio optimization problem based on the classical Markowitz framework, adapting the risk-constrained settings of Elmachtoub & Grigas (2017); Gupta & Huang (2024). Unlike shortest path, the feasible region is continuous, so market scenarios need not share the same individualized optimum; this tests whether DFC remains effective when shared decisions represent instances with different optimal portfolios. We treat each market scenario as one instance, represented by a scenario-specific return vector, with scenarios in the same cluster sharing a common portfolio decision. Full portfolio optimization formulation details are provided in Appendix E.

Data generation and experimental design. We generate 3,000 market scenarios from the synthetic process in Appendix E.1, using 2,000 for training and 1,000 as a fixed holdout set. From the training sample, we construct nested sets with $N_{\text{train}} \in \{500, 1000, 2000\}$, evaluate $K \in \{10, 20, 35, 50, 80\}$, and additionally include $N_{\text{train}} = 100$ for the fixed-$K = 10$ sample-size analysis. Each (N_{train}, K) uses 10 matched trials with the same k-means-based initialization. We report average decision loss on the training and holdout sets.

Effect of compression. DFC achieves lower holdout decision loss than k-means for every tested (N_{train}, K) combination, with larger relative reductions under stronger compression. For $N_{\text{train}} = 2000$, the reduction decreases from 8.3% at $K = 10$, where each shared portfolio represents 200 scenarios on average, to 2.9% at $K = 80$, where it represents 25. This pattern also appears for $N_{\text{train}} \in \{500, 1000\}$. Thus, when each shared decision must serve more scenarios, whether

grouped scenarios can effectively share that decision becomes more consequential. Corresponding training and holdout loss trajectories across K are reported in Appendix Figure 7.

Table 1: Portfolio optimization results on the same-distribution holdout set ($N_{\text{test}} = 1000$), averaged over 10 trials. Reduction is $100(L_{k\text{-means}} - L_{\text{DFC}})/L_{k\text{-means}}$.

N_{train}	K	N_{train}/K	Mean holdout decision loss ($\times 10^{-3}$)		Reduction (%)
			DFC	k-means	
500	10	50.00	7.453	8.006	6.9
500	20	25.00	6.119	6.453	5.2
500	35	14.29	5.433	5.659	4.0
500	50	10.00	5.103	5.256	2.9
500	80	6.25	4.803	4.870	1.4
1000	10	100.00	7.224	7.873	8.2
1000	20	50.00	5.948	6.332	6.1
1000	35	28.57	5.240	5.453	3.9
1000	50	20.00	4.877	5.040	3.2
1000	80	12.50	4.539	4.653	2.5
2000	10	200.00	7.157	7.802	**8.3**
2000	20	100.00	5.883	6.221	5.4
2000	35	57.14	5.088	5.331	4.6
2000	50	40.00	4.742	4.919	3.6
2000	80	25.00	4.369	4.497	2.9

Effect of training sample size. We next examine how DFC behaves as the amount of training data increases while holding the number of shared decisions fixed. We set $K = 10$ and evaluate all learned solutions on the same holdout set. Since the population-optimal DFC solution is unavailable, we train DFC on an independent sample of 50,000 scenarios from the same distribution and use its holdout loss as a large-sample numerical reference, as shown in Figure 5. For DFC, the difference decreases steadily with the training sample size, with the median gap falling from approximately 0.9×10^{-3} at $N_{\text{train}} = 100$ to 0.2×10^{-3} at $N_{\text{train}} = 2000$, while the corresponding gap for k-means remains larger. Thus, as more training scenarios become available, the finite-sample DFC solution approaches the performance of the large-sample DFC reference on the same holdout set. This pattern is consistent with the fixed-K statistical convergence result in Section 4. Because the 50,000-scenario solution is only a numerical reference rather than the population optimum, the experiment does not directly measure theoretical excess risk.

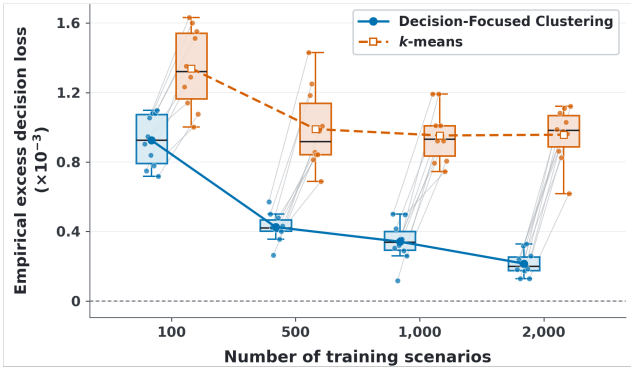


Figure 5: Holdout decision-loss gap relative to a large-sample DFC reference at fixed $K = 10$. Boxes show the interquartile range, black lines the median, and gray lines matched DFC and k-means results over 10 training samples.

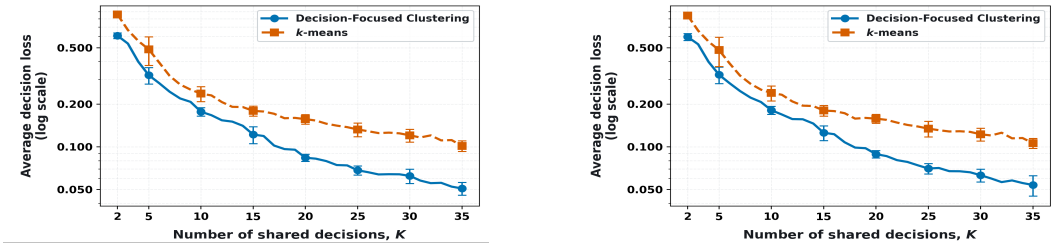
5.3 Real-World Study: Meituan Courier-Dispatching Experiment

We evaluate DFC on real-world food-delivery data from the Meituan-INFORMS-TSL Research Challenge (Meituan Group, 2024). Each courier is represented by a completed 6×5 delivery-time profile, and couriers within a cluster share the assignment induced by the same representative. Missing entries are completed using a global baseline with courier-specific region effects, yielding 4,110 profiles. Full preprocessing details are provided in Appendix F.

Decision problems. We formulate the region-level assignment problem as a continuous transportation model. Let $q \in \mathbb{R}_+^R$ denote customer-region demand and $\theta \in \mathbb{R}^{R\times M}$ the delivery-time matrix, where q_r is the observed demand in region r and θ_{rm} is the delivery time from merchant region m to customer region r. We identify θ with its vectorization in $\mathbb{R}^{RM}$ when applying DFC. For courier n, setting $\theta = \hat{\theta}^n$ gives its individualized optimal assignment, while μ_j induces the shared assignment for cluster j. Let $z \in \mathbb{R}_+^M$ denote merchant-region service capacities, with z_m estimated from the number of observed couriers and the median orders served per courier in region m. The nonnegative decision variable $x_{rm} \geq 0$ denotes the amount of demand from customer region r assigned to merchant region m. Because $x_{rm} \geq 0$ is continuous, demand from one customer region may be split across multiple merchant regions. The model minimizes aggregate delivery time subject to merchant-region service capacities and full allocation of customer-region demand.

$$\begin{aligned} &\min_{x} \sum_{r=1}^{R}\sum_{m=1}^{M} \theta_{rm}x_{rm} \\ &\text{s.t.} \sum_{r=1}^{R} x_{rm} \leq z_m \ (m=1,\dots,M), \qquad \sum_{m=1}^{M} x_{rm} = q_r \ (r=1,\dots,R). \end{aligned} \tag{4}$$

Experimental results. We randomly split the 4,110 courier profiles into 3,288 training instances and 822 testing instances. For each $K \in \{2,\dots,35\}$, we run 10 matched trials and report the average training and testing decision loss. DFC achieves lower decision loss than k-means for all tested K on both training and testing couriers, with average relative reductions of 38.0% and 37.1%, respectively. The similar reductions on unseen couriers indicate that the advantage learned from the training data persists in this real-world setting. Detailed results are reported in Appendix G.



(a) Training set.

(b) Testing set.

Figure 6: Average decision loss of DFC and k-means in the Meituan courier-dispatching experiment. Error bars show ± 1 standard deviation over 10 matched trials.

6 Conclusion

We propose Decision-Focused Clustering (DFC), motivated by the observation that geometric similarity need not imply instances can share a high-quality decision in the predict-then-optimize framework. DFC groups instances according to the decision loss incurred by sharing a decision. We develop an alternating optimization algorithm and establish statistical guarantees for exact empirical minimizers. Numerical experiments demonstrate the effectiveness of DFC in preserving downstream decision quality.

REFERENCES

Akshay Agrawal, Brandon Amos, Shane Barratt, Stephen Boyd, Steven Diamond, and J. Zico Kolter. Differentiable convex optimization layers. In *Advances in Neural Information Processing Systems*, volume 32, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/9ce3c52fc54362e22053399d3181c638-Abstract.html.

Brandon Amos and J Zico Kolter. Optnet: Differentiable optimization as a layer in neural networks. In *International conference on machine learning*, pp. 136–145. PMLR, 2017.

Arindam Banerjee, Srujana Merugu, Inderjit S Dhillon, and Joydeep Ghosh. Clustering with bregman divergences. *Journal of machine learning research*, 6(Oct):1705–1749, 2005.

Priya Donti, Brandon Amos, and J Zico Kolter. Task-based end-to-end model learning in stochastic optimization. *Advances in neural information processing systems*, 30, 2017.

Adam N. Elmachtoub and Paul Grigas. Smart "predict, then optimize". *arXiv preprint arXiv:1710.08005*, 2017.

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

Can Er and Mo Liu. Decision-focused bias correction for fluid approximation. *arXiv preprint arXiv:2512.15726*, 2025.

Aaron Ferber, Bryan Wilder, Bistra Dilkina, and Milind Tambe. MIPaaL: Mixed integer program as a layer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 1504–1511, 2020. doi: 10.1609/aaai.v34i02.5509.

M. R. Garey, D. S. Johnson, and L. Stockmeyer. Some simplified NP-complete graph problems. *Theoretical Computer Science*, 1(3):237–267, 1976. doi: 10.1016/0304-3975(76)90059-1.

Vishal Gupta and Michael Huang. Decision-focused learning with directional gradients. *Advances in Neural Information Processing Systems*, 37:79194–79220, 2024.

Wassily Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, 1963. doi: 10.1080/01621459.1963.10500830.

Xinyi Hu, Jasper Lee, and Jimmy Lee. Two-stage predict+ optimize for milps with unknown parameters in constraints. *Advances in neural information processing systems*, 36:14247–14272, 2023.

Junhyeong Lee, Sangjin Jin, and Yongjae Lee. Decision-focused learning via tangent-space projection of prediction error. *arXiv preprint arXiv:2605.01361*, 2026.

Chengzhang Li, Zhenkang Peng, and Ying Rong. Mostly beneficial clustering: Aggregating data for operational decision making. *arXiv preprint arXiv:2311.17326*, 2023.

Heyuan Liu and Paul Grigas. Risk bounds and calibration for a smart predict-then-optimize method. *Advances in Neural Information Processing Systems*, 34, 2021.

Mo Liu, Paul Grigas, Heyuan Liu, and Zuo-Jun Max Shen. Active learning in the predict-then-optimize framework: A margin-based approach. *arXiv preprint arXiv:2305.06584*, 2023.

S. Lloyd. Least squares quantization in pcm. *IEEE Transactions on Information Theory*, 28(2): 129–137, 1982. doi: 10.1109/TIT.1982.1056489.

Jayanta Mandi and Tias Guns. Interior point solving for LP-based prediction+optimisation. In *Advances in Neural Information Processing Systems*, volume 33, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/51311013e51adebc3c34d2cc591fefee-Abstract.html.

Jayanta Mandi, Emir Demirović, Peter J. Stuckey, and Tias Guns. Smart predict-and-optimize for hard combinatorial optimization problems. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 1603–1610, 2020. doi: 10.1609/aaai.v34i02.5521.

Jayanta Mandi, James Kotary, Senne Berden, Maxime Mulamba, Victor Bucarey, Tias Guns, and Ferdinando Fioretto. Decision-focused learning: Foundations, state of the art, benchmark and future opportunities. *Journal of Artificial Intelligence Research*, 81:1623–1701, 2024. doi: 10.1613/jair.1.15320.

Meituan Group. Meituan-informs-tsl research challenge dataset. `https://github.com/meituan/Meituan-INFORMS-TSL-Research-Challenge`, 2024. Data repository licensed under CC BY-NC 4.0. Accessed October 2025.

Utsav Sadana, Abhilash Chenreddy, Erick Delage, Alexandre Forel, Emma Frejinger, and Thibaut Vidal. A survey of contextual optimization methods for decision-making under uncertainty. *European Journal of Operational Research*, 320(2):271–289, 2025. doi: 10.1016/j.ejor.2024.03.020.

Beichen Wan and Mo Liu. A solver-free training method for predict-then-optimize. *arXiv preprint arXiv:2606.19587*, 2026.

Beichen Wan, Mo Liu, Paul Grigas, and Zuo-Jun Shen. Decision-focused sequential experimental design: A directional uncertainty-guided approach. In *NeurIPS 2025 Workshop MLxOR: Mathematical Foundations and Operational Integration of Machine Learning for Uncertainty-Aware Decision-Making*, 2025. URL `https://openreview.net/forum?id=d0YZKGB649`.

Bryan Wilder, Bistra Dilkina, and Milind Tambe. Melding the data-decisions pipeline: Decision-focused learning for combinatorial optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 1658–1665, 2019. doi: 10.1609/aaai.v33i01.33011658.

A Proof of Results for Computational Analysis

Proof of Theorem 1. For a cost vector c, write

$$z^\star(c) = \min_{w \in S} c^\top w, \qquad w^\star(c) \in \arg\min_{w \in S} c^\top w.$$

The decision loss is

$$\ell_{\text{decision}}(\mu, c) = c^\top w^\star(\mu) - z^\star(c).$$

We give a polynomial-time reduction from the decision version of MAX-CUT, which is NP-complete (Garey et al., 1976). Given an undirected graph $G = (V, E)$ and an integer B, this problem asks whether V can be partitioned into two sets with at least B edges between them.

Let $V = \{1, \dots, N\}$ and $d = |E|$. Orient each edge arbitrarily and write the resulting ordered endpoints as (u_r, v_r), for $r = 1, \dots, d$. Construct an instance of equation 2 with $K = 2$, downstream feasible region $S = [0, 1]^d$, and one cost vector c_i for each vertex $i \in V$, defined by

$$(c_i)_r = \begin{cases} 1, & i = u_r, \\ -1, & i = v_r, \\ 0, & \text{otherwise.} \end{cases}$$

Since optimization over S separates across coordinates,

$$z^\star(c_i) = \sum_{r=1}^{d} \min\{(c_i)_r, 0\}.$$

Each coordinate has exactly one entry equal to -1 across the cost vectors, so

$$\sum_{i=1}^{N} z^\star(c_i) = -d.$$

Consider any feasible assignment matrix $\mathbf{\Gamma}$. Let $a(i) \in \{1, 2\}$ denote the cluster assigned to instance i, and write $x_j = w^\star(\mu_j) \in [0, 1]^d$ for its shared decision. The total decision loss is

$$\begin{aligned} \sum_{i=1}^{N} \ell_{\text{decision}}(\mu_{a(i)}, c_i) &= d + \sum_{i=1}^{N} c_i^\top x_{a(i)} \\ &= \sum_{r=1}^{d} \left[1 + (x_{a(u_r)})_r - (x_{a(v_r)})_r\right]. \end{aligned} \tag{5}$$

If u_r and v_r belong to the same cluster, the corresponding summand in equation 5 equals 1. If they belong to different clusters, the summand is nonnegative, because both decision coordinates lie in $[0, 1]$. Consequently, the total decision loss is at least the number of edges whose endpoints belong to the same cluster.

This lower bound is attainable for every assignment. For an edge whose endpoints belong to different clusters, choose

$$(x_{a(u_r)})_r = 0, \qquad (x_{a(v_r)})_r = 1.$$

For an edge whose endpoints belong to the same cluster, choose $(x_1)_r = (x_2)_r = 0$. These choices can be made independently across coordinates and yield $x_1, x_2 \in \{0, 1\}^d$. Moreover, they are induced by valid cluster centers: setting

$$\mu_j = \mathbf{1} - 2x_j, \qquad j = 1, 2,$$

makes x_j the unique minimizer of $\mu_j^\top w$ over $w \in [0, 1]^d$. Thus, these decisions are attainable regardless of the optimizer's tie-breaking rule.

Let

$$C = \{i \in V : a(i) = 1\}, \qquad \delta(C) = \{\{u, v\} \in E : u \in C,\ v \notin C\}.$$

It follows that the minimum total decision loss for this assignment is exactly $d - |\delta(C)|$. Therefore,

$$\min_{\mu_1,\mu_2,\mathbf{\Gamma}} \sum_{i=1}^{N} \sum_{j=1}^{2} \gamma_{ij} \ell_{\text{decision}}(\mu_j, c_i) = d - \max_{C \subseteq V} |\delta(C)|.$$

In particular, the constructed clustering instance has optimal value at most $d - B$ if and only if G has a cut containing at least B edges.

The construction has polynomial size and can be performed in polynomial time. Hence, solving equation 2 would solve MAX-CUT, proving the claimed NP-hardness. □

Proof of Proposition 1. For a fixed cluster $\mathcal{C}_j$, $\sum_{i \in \mathcal{C}_j} \ell_{\text{decision}}(\mu, c_i) = \sum_{i \in \mathcal{C}_j} c_i^\top \big(w^*(\mu) - w^*(c_i)\big)$. The second term does not depend on μ, so minimizing over μ is equivalent to minimizing $\left(\sum_{i \in \mathcal{C}_j} c_i\right)^\top w^*(\mu) = |\mathcal{C}_j|\, \bar{c}_j^\top w^*(\mu)$. By definition, $w^*(\bar{c}_j)$ minimizes $\bar{c}_j^\top w$ over $w \in S$. Therefore, setting $\mu = \bar{c}_j$ minimizes the cluster objective. Any other representative μ satisfying $w^*(\mu) = w^*(\bar{c}_j)$ attains the same objective value. □

Proof of Theorem 2. We show that both the assignment and representative-update steps of Algorithm 1 do not increase the DFC objective.

First, consider the assignment step. For fixed representatives $\boldsymbol{\mu}^t$, Algorithm 1 assigns each instance c_i to a cluster satisfying

$$j_i^{t+1} \in \arg\min_{j=1,\dots,K} \ell_{\text{decision}}(\mu_j^t, c_i).$$

Therefore,

$$\ell_{\text{decision}}(\mu_{j_i^{t+1}}^t, c_i) \le \ell_{\text{decision}}(\mu_{j_i^t}^t, c_i)$$

for every i, and hence

$$\mathcal{L}_{\text{DFC}}\big(\mathbf{\Gamma}^{t+1}, \boldsymbol{\mu}^t\big) \le \mathcal{L}_{\text{DFC}}\big(\mathbf{\Gamma}^t, \boldsymbol{\mu}^t\big).$$

Next, consider the representative-update step. For each nonempty cluster

$$\mathcal{C}_j^{t+1} = \{i : \gamma_{ij}^{t+1} = 1\},$$

Proposition 1 shows that the update

$$\mu_j^{t+1} = \frac{1}{|\mathcal{C}_j^{t+1}|} \sum_{i \in \mathcal{C}_j^{t+1}} c_i$$

minimizes the within-cluster objective for the fixed assignments $\mathbf{\Gamma}^{t+1}$.

If $\mathcal{C}_j^{t+1} = \varnothing$, then $\gamma_{ij}^{t+1} = 0$ for every i. Hence, cluster j contributes zero to the objective under the fixed assignments $\mathbf{\Gamma}^{t+1}$, regardless of its representative. Therefore, reinitializing μ_j^{t+1} does not change the objective at this step.

Summing over all clusters yields

$$\mathcal{L}_{\text{DFC}}\big(\mathbf{\Gamma}^{t+1}, \boldsymbol{\mu}^{t+1}\big) \le \mathcal{L}_{\text{DFC}}\big(\mathbf{\Gamma}^{t+1}, \boldsymbol{\mu}^t\big).$$

Combining the two inequalities gives

$$\mathcal{L}_{\text{DFC}}\big(\mathbf{\Gamma}^{t+1}, \boldsymbol{\mu}^{t+1}\big) \le \mathcal{L}_{\text{DFC}}\big(\mathbf{\Gamma}^t, \boldsymbol{\mu}^t\big).$$

Finally, the DFC objective is nonnegative. For every cost vector c_i and representative μ_j, the optimality of $w^*(c_i)$ under cost vector c_i gives

$$c_i^\top w^*(c_i) \le c_i^\top w^*(\mu_j).$$

Therefore,

$$\ell_{\text{decision}}(\mu_j, c_i) = c_i^\top \big(w^*(\mu_j) - w^*(c_i)\big) \ge 0.$$

Hence,

$$\mathcal{L}_{\text{DFC}}(\mathbf{\Gamma}, \boldsymbol{\mu}) \ge 0.$$

The sequence of DFC objective values is therefore nonincreasing and bounded below by zero, and thus converges to a finite limit.

□

B Proofs of the Statistical Guarantees

Write

$$\widehat{R}_N(\boldsymbol{\mu}) := \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}(\boldsymbol{\mu}, c_i).$$

Compactness and loss bounds. Under either theorem, S is nonempty and compact. To verify this under Assumption 1, note that strong convexity makes g coercive, so it has a unique minimizer w_0. Moreover,

$$g(w) \geq g_{\min} + \frac{\sigma}{2}\|w - w_0\|_2^2.$$

Hence, every $w \in S$ satisfies $\|w - w_0\|_2 \leq \sqrt{2(r - g_{\min})/\sigma}$. Continuity and convexity of g imply that S is closed and convex, and $r > g_{\min}$ ensures that it is nonempty. Thus $D < \infty$. If $D = 0$, all feasible decisions coincide and both theorems are immediate; henceforth assume $D > 0$.

Optimality of $w^*(c)$ implies

$$0 \leq \mathcal{L}(\boldsymbol{\mu}, c) \leq D\|c\|_2. \tag{6}$$

For any representative tuples $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$,

$$\big|\mathcal{L}(\boldsymbol{\mu}, c) - \mathcal{L}(\boldsymbol{\nu}, c)\big| \leq \|c\|_2 \max_{j\in[K]} \|w^*(\mu_j) - w^*(\nu_j)\|_2. \tag{7}$$

Indeed, the individualized optimal cost cancels, and $|\min_j a_j - \min_j b_j| \leq \max_j |a_j - b_j|$. Thus all losses are integrable, and closeness of the induced decisions controls their loss differences without requiring bounded representatives.

Attainment and empirical optimality. The population minimum is attained. To see this, first minimize

$$\mathbb{E}_{c\sim P}\left[\min_{j\in[K]} c^\top w_j - c^\top w^*(c)\right]$$

over arbitrary shared decisions $(w_1, \ldots, w_K) \in S^K$. The same Lipschitz estimate as in equation 7, together with integrability, makes this objective continuous. Compactness therefore ensures attainment. Assign each instance to its lowest-cost decision, breaking ties by the smallest index. For each positive-probability cell A_j, replace its decision by $w^*(\mathbb{E}[c \mid c \in A_j])$; use arbitrary representatives for zero-probability cells. These replacements cannot increase expected cost under the old assignments, and reassignment cannot increase it either. The resulting representative tuple therefore attains the same minimum. The argument also applies to the empirical distribution.

Define

$$U_N := \sup_{\boldsymbol{\mu}\in(\mathbb{R}^d)^K} \big|\widehat{R}_N(\boldsymbol{\mu}) - R(\boldsymbol{\mu})\big|.$$

For any $\boldsymbol{\mu}$, exact empirical minimization gives

$$\begin{aligned} R(\widehat{\boldsymbol{\mu}}_N) &\leq \widehat{R}_N(\widehat{\boldsymbol{\mu}}_N) + U_N \\ &\leq \widehat{R}_N(\boldsymbol{\mu}) + U_N \leq R(\boldsymbol{\mu}) + 2U_N. \end{aligned}$$

Taking the minimum over $\boldsymbol{\mu}$ yields

$$0 \leq R(\widehat{\boldsymbol{\mu}}_N) - R^* \leq 2U_N. \tag{8}$$

Proof of Theorem 3. Because the oracle selects vertices, every representative tuple induces an element of V^K. Choose a finite set $\mathcal{M}_V \subseteq (\mathbb{R}^d)^K$ containing one representative tuple for each attainable tuple of shared decisions. Then $|\mathcal{M}_V| \leq M^K$, and every loss function $\mathcal{L}(\boldsymbol{\mu}, \cdot)$ equals one indexed by $\mathcal{M}_V$. Consequently,

$$U_N = \max_{\boldsymbol{\mu}\in\mathcal{M}_V} \big|\widehat{R}_N(\boldsymbol{\mu}) - R(\boldsymbol{\mu})\big|.$$

Each of these finitely many losses is integrable by equation 6. The strong law of large numbers therefore gives $U_N \to 0$ almost surely.

If every tuple in $\mathcal{M}_V$ is population-optimal, the conclusion holds for every N. Otherwise, finiteness implies

$$\Delta := \min_{\substack{\boldsymbol{\mu}\in\mathcal{M}_V \\ R(\boldsymbol{\mu})>R^*}} \left\{R(\boldsymbol{\mu}) - R^*\right\} > 0.$$

Almost surely, there exists a finite random N_0 such that $2U_N < \Delta$ for all $N \geq N_0$. By equation 8, the excess risk of $\widehat{\boldsymbol{\mu}}_N$ is then strictly smaller than Δ. Its induced decisions agree with those of a tuple in $\mathcal{M}_V$, whose excess risk is either zero or at least Δ. Hence,

$$R(\widehat{\boldsymbol{\mu}}_N) - R^* = 0 \qquad \text{for every } N \geq N_0,$$

proving equation 3. □

Proof of Theorem 4. Suppose $\|c\|_2 \leq B$ almost surely. If $B = 0$, all losses vanish; otherwise assume $B > 0$.

Fix $\eta > 0$. Choose a maximal subset of the attainable decisions $\{w^*(\mu) : \mu \in \mathbb{R}^d\} \subseteq S$ whose distinct elements are more than η apart. Choose one representative cost vector for each selected decision and denote the resulting set by $\mathcal{Q}_\eta$. The open balls of radius $\eta/2$ centered at the selected decisions are disjoint and lie in a ball of radius $D + \eta/2$ centered at any one of them. Comparing volumes shows that the set is finite and

$$|\mathcal{Q}_\eta| \leq (1 + 2D/\eta)^d.$$

Maximality ensures that every attainable decision is within distance η of a selected decision. Therefore, $\mathcal{N}_\eta := \mathcal{Q}_\eta^K$ has at most $(1 + 2D/\eta)^{Kd}$ elements, and every $\boldsymbol{\mu}$ has an approximation $\boldsymbol{\nu} \in \mathcal{N}_\eta$ satisfying

$$\max_{j\in[K]} \|w^*(\mu_j) - w^*(\nu_j)\|_2 \leq \eta.$$

By equation 7, their losses differ by at most $B\eta$ almost surely. Applying this estimate to both empirical and population risks yields

$$U_N \leq \max_{\boldsymbol{\nu}\in\mathcal{N}_\eta} \left|\widehat{R}_N(\boldsymbol{\nu}) - R(\boldsymbol{\nu})\right| + 2B\eta. \tag{9}$$

For each $\boldsymbol{\nu} \in \mathcal{N}_\eta$, equation 6 gives $0 \leq \mathcal{L}(\boldsymbol{\nu}, c) \leq BD$ almost surely. Hoeffding's inequality (Hoeffding, 1963) and a union bound imply that, for every $t > 0$,

$$\mathbb{P}\left\{\max_{\boldsymbol{\nu}\in\mathcal{N}_\eta} \left|\widehat{R}_N(\boldsymbol{\nu}) - R(\boldsymbol{\nu})\right| > t\right\} \leq 2(1 + 2D/\eta)^{Kd} \exp\left(-\frac{2Nt^2}{B^2D^2}\right).$$

Thus, with probability at least $1 - \delta$,

$$U_N \leq BD\sqrt{\frac{Kd\log(1 + 2D/\eta) + \log(2/\delta)}{2N}} + 2B\eta.$$

Combining this with equation 8 and setting $\eta = D/N$ gives

$$R(\widehat{\boldsymbol{\mu}}_N) - R^* \leq BD\sqrt{\frac{2\{Kd\log(1 + 2N) + \log(2/\delta)\}}{N}} + \frac{4BD}{N},$$

which proves Theorem 4. The argument uses compactness but does not require uniqueness of the population optimum or smoothness of the map from representatives to decisions. □

B.1 Consequences for Fixed Population Clusters

The preceding theorems concern jointly learned clusters under exact empirical minimization. For a fixed measurable population cluster A_j^*, write $p_j = P\{c \in A_j^*\} > 0$ and $\mu_j^* = \mathbb{E}[c \mid c \in A_j^*]$. Its mean can be analyzed directly without a global-optimization assumption. Define

$$n_j(N) = \sum_{i=1}^{N} \mathbf{1}\{c_i \in A_j^*\}, \qquad \overline{c}_{j,N} = \frac{\sum_{i=1}^{N} c_i \mathbf{1}\{c_i \in A_j^*\}}{n_j(N)}$$

whenever $n_j(N) > 0$. The strong law gives

$$\frac{n_j(N)}{N} \to p_j, \qquad \frac{1}{N}\sum_{i=1}^{N} c_i \mathbf{1}\{c_i \in A_j^*\} \to p_j \mu_j^*,$$

and hence $\overline{c}_{j,N} \to \mu_j^*$ almost surely.

For a polytope, write $z^*(m) = m^\top w^*(m)$ and let $V_j^* = \{v \in V : (\mu_j^*)^\top v = z^*(\mu_j^*)\}$. If $V_j^* = V$, every feasible decision is optimal at μ_j^* and the regret is identically zero. Otherwise set

$$\Delta_j = \min_{v \in V \setminus V_j^*} \{(\mu_j^*)^\top v - z^*(\mu_j^*)\} > 0.$$

For any $v_0 \in V_j^*$ and $v \notin V_j^*$,

$$\overline{c}_{j,N}^\top (v - v_0) \geq \Delta_j - D\|\overline{c}_{j,N} - \mu_j^*\|_2.$$

Once the norm error is smaller than Δ_j / D, no vertex outside V_j^* can be optimal under $\overline{c}_{j,N}$. Every optimizer therefore lies in $\mathrm{conv}(V_j^*)$, and

$$\ell_{\mathrm{decision}}(\overline{c}_{j,N}, \mu_j^*) = 0 \quad \text{for all sufficiently large } N \quad \text{almost surely.}$$

This fixed-cluster conclusion does not require a unique population optimizer or a vertex-valued oracle.

For an α-strongly convex set, the following bound is useful:

$$0 \leq \ell_{\mathrm{decision}}(\mu, m) \leq \frac{2}{\alpha\|m\|_2}\|\mu - m\|_2^2, \qquad m \neq 0. \tag{10}$$

To see this, write $w_m = w^*(m)$. For $w \in S$ and $t \in (0,1)$, strong convexity gives

$$(1-t)w_m + tw - \frac{\alpha}{2}t(1-t)\|w - w_m\|_2^2 \frac{m}{\|m\|_2} \in S.$$

Optimality of w_m, division by t, and the limit $t \downarrow 0$ yield

$$m^\top (w - w_m) \geq \frac{\alpha}{2}\|m\|_2 \|w - w_m\|_2^2. \tag{11}$$

Set $\widehat{w} = w^*(\mu)$ and $q = \|\widehat{w} - w_m\|_2$. Since $\mu^\top(\widehat{w} - w_m) \leq 0$,

$$\frac{\alpha}{2}\|m\|_2 q^2 \leq \ell_{\mathrm{decision}}(\mu, m) \leq (m - \mu)^\top (\widehat{w} - w_m) \leq \|m - \mu\|_2 q.$$

When $q > 0$, this gives $q \leq 2\|m - \mu\|_2 / (\alpha\|m\|_2)$, proving equation 10; when $q = 0$, the bound is immediate.

For an α-strongly convex set and $\mu_j^* \neq 0$, equation 10 gives almost-sure convergence of the same regret to zero. If the conditional covariance Σ_j is finite, and $\overline{c}_{j,n}$ is the mean of n independent observations from $P(\cdot \mid A_j^*)$, then

$$\mathbb{E}[\ell_{\mathrm{decision}}(\overline{c}_{j,n}, \mu_j^*)] \leq \frac{2\,\mathrm{tr}(\Sigma_j)}{\alpha\|\mu_j^*\|_2\, n}. \tag{12}$$

Indeed, the mean-squared norm error is $\mathrm{tr}(\Sigma_j)/n$, and taking expectations in equation 10 proves the bound. If $\mu_j^* = 0$, the conditional excess regret is zero for every representative. The $O(n^{-1})$ bound concerns mean estimation within a fixed cluster; it is not a global $O(N^{-1})$ rate for learning both assignments and representatives.

Why finite identification need not hold for a strongly convex set. Take $K = 1$, $S = \{w \in \mathbb{R}^2 : \|w\|_2 \leq 1\}$, and $c_i = (1, Z_i)$ with Z_i independent and uniform on $[-1, 1]$. The population mean is $\mu^* = (1, 0)$, and the empirical mean is $\widehat{\mu}_N = (1, \overline{Z}_N)$. Since $w^*(c) = -c/\|c\|_2$ for $c \neq 0$,

$$\ell_{\mathrm{decision}}(\widehat{\mu}_N, \mu^*) = 1 - \frac{1}{\sqrt{1 + \overline{Z}_N^2}}.$$

For each finite N, $\overline{Z}_N$ has a continuous distribution, so this regret is strictly positive almost surely. Taking a countable intersection, it is almost surely positive for every finite N, while it converges to zero by the strong law. This is already an exact empirical DFC solution with $K = 1$, so the absence of finite identification is not an optimization artifact.

C DECISION CONSISTENCY VERSUS PARAMETER CONSISTENCY.

Decision loss does not identify a unique representative cost vector. With a tie-breaking rule that depends only on the optimal face, positive rescaling preserves the decision: $w^*(a\mu) = w^*(\mu)$ for every $a > 0$. The same equality holds without an additional tie-breaking assumption whenever the optimizer is unique. Thus, for any nonzero population representative μ_j^* and $a_N \to \infty$, the decision-equivalent representatives $\widetilde{\mu}_{j,N} = a_N \mu_j^*$ satisfy $\ell_{\text{decision}}(\widetilde{\mu}_{j,N}, \mu_j^*) = 0$, $\|\widetilde{\mu}_{j,N} - \mu_j^*\|_2 = |a_N - 1| \, \|\mu_j^*\|_2 \to \infty$. This phenomenon applies to both polyhedral and strongly convex feasible regions. It establishes that decision-loss consistency does not imply Euclidean consistency of unrestricted representatives. It does not imply that the empirical means used in Proposition 1 diverge: for a fixed positive-probability cluster, empirical means converge almost surely under the integrability assumption, and bounded data yield bounded empirical cluster means.

D SHORTEST-PATH FORMULATION AND ADDITIONAL RESULTS

Consider a directed 3×3 grid graph with source node s (top-left) and sink node t (bottom-right). Let E denote the set of directed edges and V the set of nodes. The grid contains 12 directed edges, corresponding to the horizontal and vertical moves toward the sink. Each synthetic instance is represented by a cost vector $c \in \mathbb{R}^{12}$, with one component c_e specifying the cost of each edge $e \in E$. The binary decision variable $x_e \in \{0, 1\}$ indicates whether the edge e is included in the path. The shortest-path problem is formulated as follows:

$$\min_{x \in \{0,1\}^{|E|}} c^\top x \quad \text{s.t.} \quad \sum_{e \in \delta^+(v)} x_e - \sum_{e \in \delta^-(v)} x_e = b_v \tag{13}$$

The flow-balance constraint holds for all $v \in V$, where

$$b_v = \begin{cases} 1 & v = s \text{ (source)}, \\ -1 & v = t \text{ (sink)}, \\ 0 & \text{otherwise.} \end{cases}$$

Here, $\delta^+(v)$ and $\delta^-(v)$ denote the sets of edges leaving and entering node v, respectively. Solving equation 13 returns the shortest-path decision $w^*(c)$ for any cost vector c, which serves as the downstream decision throughout the experiment. Under DFC, each representative μ_j determines the shared path $w^*(\mu_j)$ for the instances assigned to cluster j.

D.1 DETAILED SHORTEST-PATH RESULTS

Table 2 reports the average decision loss per instance over 35 trials for each value of K. Relative reduction is computed as

$$\frac{L_{k\text{-means}} - L_{\text{DFC}}}{L_{k\text{-means}}} \times 100\%.$$

Table 2: Average decision loss per instance and relative reduction of DFC over k-means on the shortest-path experiment. Results are averaged over 35 trials.

	Training			Testing		
K	DFC	k-means	Reduction (%)	DFC	k-means	Reduction (%)
3	0.303745	0.651770	53.40%	0.319255	0.672507	52.53%
4	0.131736	0.433151	69.59%	0.145017	0.450016	67.78%
5	0.044017	0.263114	83.27%	0.046255	0.276967	83.30%
6	0.000546	0.226678	99.76%	0.000478	0.239885	99.80%
7	0.000000	0.217556	100.00%	0.000000	0.231895	100.00%
8	0.000000	0.185628	100.00%	0.000000	0.203929	100.00%
9	0.000000	0.153701	100.00%	0.000000	0.175963	100.00%
10	0.000000	0.117212	100.00%	0.000000	0.144002	100.00%

E PORTFOLIO FORMULATION, DATA GENERATION AND ADDITIONAL RESULTS

The decision maker allocates capital across M risky assets while controlling portfolio risk. We treat each market scenario as one instance, represented by a scenario-specific return vector. The N scenarios are partitioned into K clusters, and scenarios in the same cluster share a common portfolio decision.

For each market scenario i, let $r_i = (r_{i1}, \ldots, r_{iM}) \in \mathbb{R}^M$ denote the vector of incremental asset returns above the risk-free rate. Following the minimization convention used throughout our framework, we define the corresponding cost vector as $c_i = -r_i$. Thus, minimizing $c_i^\top w$ is equivalent to maximizing the portfolio's incremental return.

Let $w_m \geq 0$ denote the fraction of capital allocated to risky asset m, and let $w = (w_1, \ldots, w_M)$ denote the portfolio decision. All market scenarios share the same portfolio constraints, while the objective coefficients vary through the scenario-specific cost vector c. Given a cost vector c, the optimal portfolio is obtained by solving

$$
\begin{aligned}
w^*(c) \in \arg\min_{w} \quad & c^\top w \\
\text{s.t.} \quad & w^\top \Sigma w \leq \gamma, \\
& \mathbf{1}^\top w \leq 1, \\
& w \geq 0,
\end{aligned}
\tag{14}
$$

where Σ is the covariance matrix of asset returns and γ specifies the portfolio risk limit. The constraint $w^\top \Sigma w \leq \gamma$ limits the portfolio variance, $\mathbf{1}^\top w \leq 1$ allows any unallocated capital to remain in the risk-free asset, and $w \geq 0$ rules out short selling.

For cluster j, the representative cost vector μ_j induces the shared portfolio $w^*(\mu_j)$. Following equation 1, $\ell_{\text{decision}}(\mu_j, c_i)$ measures the loss incurred when scenario i uses the shared portfolio $w^*(\mu_j)$ instead of its individualized optimum $w^*(c_i)$.

E.1 DATA GENERATION

Following the experimental design of Elmachtoub & Grigas (2017), we generate synthetic portfolio instances with $M = 50$ risky assets, $p = 5$ observable market features, four latent factors, and $\tau = 1$.

We first generate a coefficient matrix $B \in \{0, 1\}^{50 \times 5}$, with entries independently sampled from a Bernoulli distribution with probability 0.5, and a factor loading matrix $L \in \mathbb{R}^{50 \times 4}$, with entries independently sampled from $\mathcal{U}(-0.0025\tau, 0.0025\tau)$. The matrix B determines how observable market features affect the conditional mean asset returns. The matrix L determines how the latent market factors contribute to variation across asset returns.

For each market scenario i, we independently generate $x_i \sim \mathcal{N}(0, I_5)$. These features determine the conditional mean return of asset j as $\bar{r}_{ij} = \left[\frac{0.05}{\sqrt{5}}(Bx_i)_j + 0.1^{1/4}\right]^4$. We then generate a latent market factor $f_i \sim \mathcal{N}(0, I_4)$ and the asset-specific noise $\epsilon_i \sim \mathcal{N}(0, I_{50})$. The realized return vector is $r_i = \bar{r}_i + Lf_i + 0.01\tau\epsilon_i$. The factor term Lf_i introduces correlated variation across asset returns, and $0.01\tau\epsilon_i$ captures asset-specific variation.

All market scenarios share the covariance matrix $\Sigma = LL^\top + (0.01\tau)^2 I_{50}$. We use the equal-weight portfolio $\bar{w} = \frac{1}{50}\mathbf{1}$ as a reference and set the risk limit to $\gamma = 2.25\, \bar{w}^\top \Sigma \bar{w}$.

The training and holdout scenarios share the same global parameters B, L, Σ, and γ, while the scenario-specific variables x_i, f_i, and ϵ_i are generated independently. Thus, the training and holdout sets are independent samples from the same underlying data-generating process.

E.2 Additional Portfolio Results

We report average decision loss per instance on both the training and holdout sets. On the training set, this is

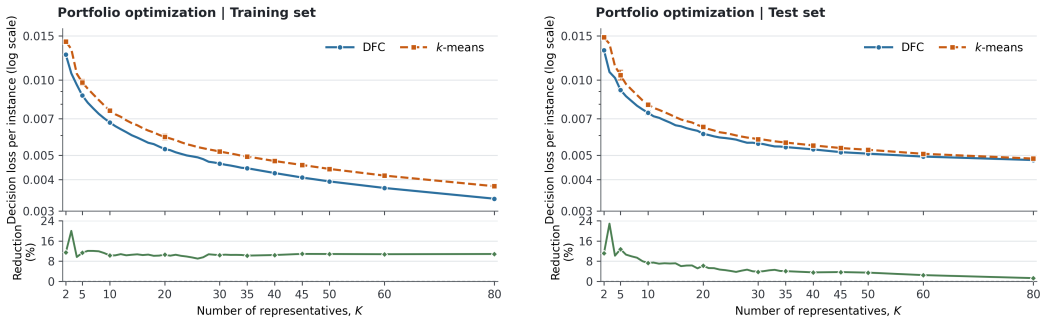
$$\frac{1}{N_{\text{train}}} \sum_{i=1}^{N_{\text{train}}} \min_{j \in [K]} \ell_{\text{decision}}(\mu_j, c_i).$$

For the holdout set, the learned representatives are held fixed, and each held-out scenario is assigned to the representative that yields the smallest decision loss. The average holdout loss is

$$\frac{1}{N_{\text{test}}} \sum_{i=1}^{N_{\text{test}}} \min_{j \in [K]} \ell_{\text{decision}}(\mu_j, c_i),$$

where $N_{\text{test}} = 1000$. The holdout scenarios are used only for evaluation and do not affect the learned representatives.

Figure 7 reports the corresponding training and holdout decision-loss trajectories across the full tested range of K. For both DFC and k-means, absolute decision loss decreases as more shared portfolio decisions are available. DFC achieves lower loss throughout the tested range on both the training and holdout sets, while the gap between the two methods becomes smaller on the holdout set as K increases. This pattern is consistent with the compression results reported in the main text, where the relative reduction of DFC is larger when fewer shared decisions are available.



(a) Training set.

(b) Holdout set.

Figure 7: Average decision loss of DFC and k-means across the full tested range of K in the portfolio experiment.

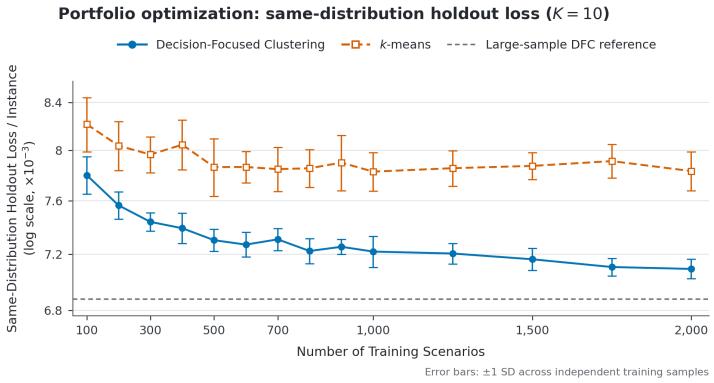


Figure 8: Same-distribution holdout decision loss as the training sample size increases at fixed $K = 10$. The dashed horizontal line denotes the DFC reference solution trained on 50,000 scenarios and evaluated on the same holdout set. Error bars show ± 1 standard deviation over independent training samples.

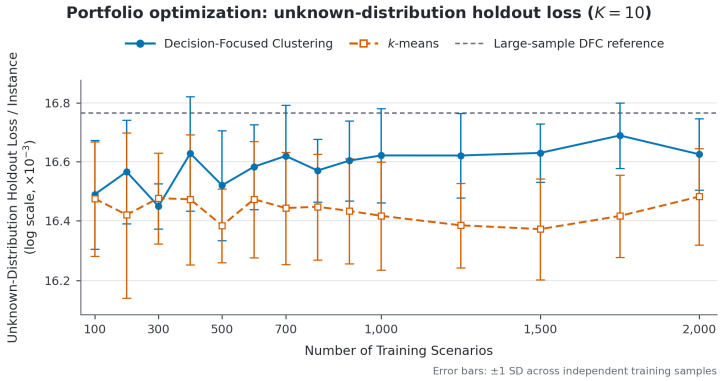


Figure 9: Holdout decision loss under distribution shift at fixed $K = 10$. Error bars show ± 1 standard deviation over independent training samples.

The current DFC framework is static: the cluster representatives are learned once from the training data and then held fixed when evaluating new instances. Thus, the same-distribution holdout experiment examines whether the learned shared decisions generalize to unseen instances drawn from the same underlying distribution. DFC is trained to optimize shared decisions for the training distribution and does not explicitly model distribution shift. Accordingly, the advantage observed under same-distribution holdout evaluation need not persist when the test distribution changes. Figure 9 illustrates this limitation: the advantage observed in the same-distribution setting does not persist under the shifted distribution, and k-means attains lower holdout decision loss across most training sample sizes. This result highlights that the current framework addresses static shared-decision learning rather than adaptation to changing distributions.

F Meituan Data Preprocessing

We use one week of anonymized food-delivery data from the Meituan-INFORMS-TSL Research Challenge, collected in October 2022 in a city in Northern China. The dataset includes 568,546 orders completed by 4,955 couriers and involving 4,962 merchants.

Customer-region identifiers are provided in the dataset, and we retain the six regions with the largest numbers of observed orders. Merchant regions are constructed by applying DBSCAN to merchant locations and retaining five major regions, with the clustering details and resulting spatial partition provided in Appendix Figure 10. This produces a 6×5 set of customer–merchant region pairs. We index customer regions by $r = 1, \ldots, R$ and merchant regions by $m = 1, \ldots, M$, with $R = 6$ and $M = 5$. Delivery time is measured in minutes from dispatch to arrival. We remove nonpositive delivery times and values above the 99.5th percentile before constructing the region-level delivery-time matrices.

We first construct a global baseline delivery-time matrix $\bar{\theta} \in \mathbb{R}^{6\times 5}$. For each observed customer–merchant region pair (r, m), $\bar{\theta}_{rm}$ is the mean delivery time across all corresponding orders. Missing entries are imputed using a linear regression model based on spatial, temporal, and operational covariates; details are provided in Appendix F.2.

For courier n, let θ^n_{rm} denote the average observed delivery time from merchant region m to customer region r. Because each courier serves only a subset of customer–merchant region pairs, many entries of θ^n are unobserved. For each observed region pair (r, m), we define the courier-specific deviation from the global baseline as $\Delta^n_{rm} = \theta^n_{rm} - \bar{\theta}_{rm}$. We approximate this deviation additively as $\Delta^n_{rm} \approx \delta^n_r + \gamma^n_m$, where δ^n_r and γ^n_m are courier-specific customer- and merchant-region effects. For an unobserved region pair, we estimate $\hat{\theta}^n_{rm} = \bar{\theta}_{rm} + \delta^n_r + \gamma^n_m$, while observed entries retain their empirical courier-specific averages. Details of the iterative estimation procedure are provided in Appendix F.3.

This procedure produces a complete delivery-time matrix $\hat{\theta}^n \in \mathbb{R}^{6\times 5}$ for each courier. After preprocessing and restricting the data to the selected customer and merchant regions, we retain 352,256

orders from 4,110 couriers. These 4,110 completed courier profiles serve as the input instances for DFC. The completed global baseline matrix used for courier-level imputation is reported in Appendix F.4.

F.1 Merchant Region Construction

Merchant regions are constructed from merchant latitude and longitude coordinates using DBSCAN with $\epsilon = 0.003$ and $\texttt{min_samples} = 5$. We retain the five largest non-noise clusters as the merchant regions. Merchants assigned to smaller clusters or identified as noise are reassigned to the nearest retained cluster center. Figure 10 shows the resulting spatial partition.

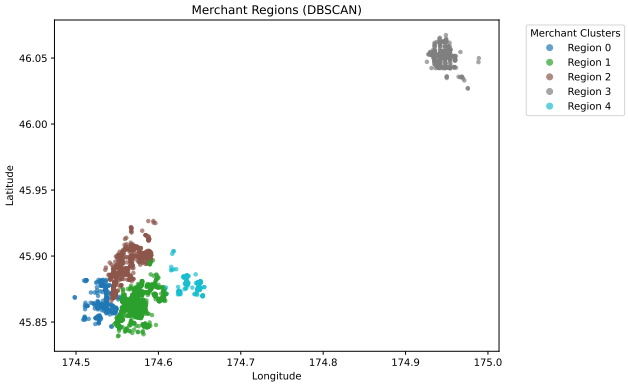


Figure 10: Merchant regions constructed using DBSCAN with $\epsilon = 0.003$ and $\texttt{min_samples} = 5$.

F.2 Global Baseline Matrix

We construct a global baseline delivery-time matrix $\bar{\theta} \in \mathbb{R}^{6\times5}$. For each observed customer–merchant region pair (r, m), $\bar{\theta}_{rm}$ is the mean delivery time across all orders assigned to that pair. For region pairs with no observed orders, we estimate the missing entries using a linear regression model trained on order-level observations from the selected customer and merchant regions.

The regression uses spatial, temporal, and operational covariates, including pickup-to-dropoff distance and its nonlinear terms, courier-to-restaurant and dispatch-to-grab times, regional order density, merchant preparation time, time-of-day and weekday indicators, peak and weekend indicators, and several interaction terms. For an unobserved customer–merchant region pair, we compute the Haversine distance between the corresponding customer-region and merchant-region centers and combine it with region-level averages of the remaining operational covariates. Temporal covariates are fixed to a representative weekday lunch-period setting. The resulting prediction is clipped to the range $[1, q_{0.99}]$, where $q_{0.99}$ is the 99th percentile of observed delivery times after preprocessing.

F.3 Courier-Specific Profile Completion

To complete the unobserved entries, we start from the global baseline $\bar{\theta}$ and define the courier-specific deviation for each observed customer–merchant region pair (r, m) as $\Delta^n_{rm} = \theta^n_{rm} - \bar{\theta}_{rm}$. We approximate this deviation using an additive customer-region and merchant-region structure, $\Delta^n_{rm} \approx \delta^n_r + \gamma^n_m$, where δ^n_r and γ^n_m are courier-specific customer- and merchant-region effects.

The customer-region and merchant-region deviations are initialized at zero and updated iteratively using the observed region pairs. For each observed pair, we compute the residual $e^n_{rm} = \Delta^n_{rm} - (\delta^n_r + \gamma^n_m)$. If $|e^n_{rm}| \geq 2$ minutes, the residual is split equally to update δ^n_r and γ^n_m. We repeat this update over the observed entries for at most two iterations. For an unobserved customer–merchant region pair (r, m), we estimate the courier-specific delivery time as $\hat{\theta}^n_{rm} = \bar{\theta}_{rm} + \delta^n_r + \gamma^n_m$. Observed entries retain their empirical courier-specific averages. Finally, all completed values are clipped to the range of the entries in the global baseline matrix.

F.4 Completed Global Baseline Delivery-Time Matrix

Table 3 reports the completed global baseline delivery-time matrix $\bar{\theta}$ used in the Meituan courier-dispatching experiment. Rows correspond to the six selected customer regions and columns correspond to the five merchant regions. All entries are measured in minutes.

Table 3: Completed global baseline delivery-time matrix $\bar{\theta}$ used for courier-level imputation.

	Merchant region				
Customer region	0	1	2	3	4
0	26.15	28.36	30.41	55.58	47.00
2	25.08	28.81	29.70	55.58	52.36
5	40.28	40.40	28.62	55.58	49.26
6	55.58	55.58	55.58	19.96	55.58
7	34.83	27.32	36.91	55.58	42.47
9	47.39	27.16	26.31	55.58	23.58

G Meituan Courier-Dispatching Experiment Results

For each K, relative reduction is computed as

$$100\frac{L_{k\text{-means},K} - L_{\mathrm{DFC},K}}{L_{k\text{-means},K}}.$$

The reported average reduction is the arithmetic mean of these relative reductions across the tested values of K.

Table 4: Average decision loss per order and relative reduction of DFC over k-means on the Meituan courier-dispatching experiment. Results are averaged over 10 trials ($N_{\text{train}} = 3288$, $N_{\text{test}} = 822$). The ratio N_{train}/K is the average number of training instances represented by each shared decision.

		Training			Testing		
K	N_{train}/K	DFC	k-means	Reduction (%)	DFC	k-means	Reduction (%)
2	1644.00	0.609	0.860	29.2%	0.597	0.845	29.3%
3	1096.00	0.536	0.668	19.8%	0.530	0.661	19.9%
4	822.00	0.399	0.558	28.6%	0.398	0.553	28.0%
5	657.60	0.321	0.488	34.3%	0.323	0.482	33.0%
6	548.00	0.282	0.394	28.5%	0.282	0.389	27.5%
7	469.71	0.245	0.317	22.7%	0.248	0.319	22.2%
8	411.00	0.220	0.276	20.2%	0.223	0.279	20.1%
9	365.33	0.208	0.254	18.1%	0.207	0.254	18.2%
10	328.80	0.177	0.237	25.4%	0.181	0.240	24.4%
11	298.91	0.167	0.231	27.5%	0.169	0.233	27.4%
12	274.00	0.154	0.207	25.6%	0.157	0.209	24.9%
13	252.92	0.151	0.192	21.3%	0.157	0.195	19.8%
14	234.86	0.141	0.191	26.2%	0.146	0.194	24.7%
15	219.20	0.122	0.180	31.9%	0.126	0.180	30.3%
16	205.50	0.119	0.178	32.8%	0.123	0.179	31.3%
17	193.41	0.102	0.171	40.3%	0.108	0.173	37.6%
18	182.67	0.097	0.159	39.4%	0.099	0.158	37.5%
19	173.05	0.096	0.160	40.4%	0.098	0.160	38.9%
20	164.40	0.084	0.157	46.6%	0.089	0.158	43.9%
21	156.57	0.083	0.153	46.2%	0.086	0.155	44.5%
22	149.45	0.079	0.146	45.5%	0.081	0.148	45.7%
23	142.96	0.075	0.141	47.3%	0.078	0.143	45.2%
24	137.00	0.074	0.138	46.3%	0.074	0.139	46.8%
25	131.52	0.068	0.133	48.5%	0.070	0.135	47.7%
26	126.46	0.066	0.129	48.6%	0.071	0.131	45.9%
27	121.78	0.064	0.125	48.9%	0.067	0.128	47.5%
28	117.43	0.064	0.126	49.1%	0.067	0.129	47.8%
29	113.38	0.064	0.125	48.6%	0.066	0.128	48.1%
30	109.60	0.062	0.121	48.2%	0.063	0.123	48.6%
31	106.06	0.058	0.117	50.3%	0.060	0.120	50.3%
32	102.75	0.056	0.121	54.0%	0.056	0.126	55.2%
33	99.64	0.056	0.111	49.8%	0.058	0.115	49.5%
34	96.71	0.052	0.112	53.0%	0.056	0.116	51.8%
35	93.94	0.051	0.102	49.9%	0.054	0.106	49.3%

AI USE STATEMENT

In this work, we used generative AI tools to aid and polish the writing of this paper and to assist in drafting sections of the manuscript. We have not used generative AI tools for literature retrieval or discovery, research ideation, generating synthetic datasets, or proving mathematical claims. Additionally, we used generative AI tools to assist with editing and debugging experimental code based on experiment designs specified by the authors. We have reviewed all AI-assisted work. The research questions, experimental designs, evaluation procedures, and interpretation of the reported results were determined and verified by the authors. All AI-assisted text was reviewed and edited for accuracy, clarity, and consistency with the underlying research, and all AI-assisted code was reviewed, tested, and verified against the intended experimental specifications. We take responsibility for the final content of this work, including text, code, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

The real-world experiment uses anonymized food-delivery data from the Meituan-INFORMS-TSL Research Challenge. We analyze aggregated courier-level delivery-time profiles and do not attempt to identify individual couriers, customers, or merchants. The data are used solely for evaluating the proposed methodology, and no personally identifiable information is released or reconstructed.

REPRODUCIBILITY STATEMENT

We provide the full formulation of Decision-Focused Clustering, the optimization procedure, and the assumptions underlying the theoretical results in the main text. Complete proofs are included in the appendix. The appendix also reports the synthetic data-generation procedures, experimental settings, and detailed preprocessing steps for the Meituan dataset. These materials provide the information needed to reproduce the theoretical results and empirical evaluations reported in this paper.