**摘要**现实数据往往同时含缺失、异常等多种错误，这种数据质量波动会放大无监督 AutoML的搜索空间并降低结果稳定性。本文提出一个核心假设：数据质量与聚类特征之间存在可预测的结构性，只要捕获足够的数据质量及算法特征，即可提前筛出高潜力的清洗–聚类组合。围绕该假设，我们构建了三条策略：离线阶段通过特征增强提高候选命中率；搜索阶段使用剪枝压缩空间至O(k log n)；在线阶段监控聚类统计量并动态修正参数漂移。实验部分按“现象→机理→价值”的流程进行，在不同规模的数据集上验证了该方案比全量搜索提高 8 倍以上的运行速度，同时聚类质量平均损失不超过 2 %。研究结果表明，所提出的协同优化框架能够在大型脏污数据场景中兼顾速度与稳健性。

# **1 引言**

在医疗精准诊疗、金融风险管理与工业物联网等数据密集型场景中，无监督聚类已成为发现潜在模式、辅助决策的重要工具。然而，聚类算法对数据分布的依赖极为敏感：一条拼写错误、一处未标记的异常或一段缺失区间，便可能扭曲距离结构和密度估计，从而打乱簇划分、收敛路径乃至下游分析链条。与依赖标签缓冲噪声的有监督学习不同，聚类几乎完全暴露在原始数据质量之下。一个直观的解决思路是借助 AutoML将若干数据清洗算子与多种聚类算法串联，交给优化器在组合空间中自动择优。然而，随着清洗算子、聚类模型与超参数维度的指数级增长，优化器往往在大量注定失败的方案上浪费算力，从而削弱了自动化本身的价值。

过去十年，数据清洗与聚类算法各自取得诸多突破：针对多源异构数据，缺失值插补、异常检测、容错匹配等数据清洗方法层出不穷；而 K-Means、DBSCAN、层次聚类乃至深度嵌入聚类，在复杂数据形态下展现出较好的可塑性。然而，**两条技术路线几乎是独立推进的**：清洗研究聚焦局部错误修复，很少衡量修复对无监督任务的连锁效应；聚类研究默许输入“足够干净”，将主要精力投入相似度度量和优化策略。当数据体量与脏污类型同步增长时，这种割裂带来了三重放大的复杂性：清洗会改变距离分布，致使既有超参数失配；跨表、跨批次的数据质量高度异质，单一策略难以通用；指数级候选组合更令系统开销飙升，无法通过增加算力线性抵消。**业界已尝试用 AutoML 串联清洗与聚类**，但在缺乏先验认知的情况下，这一做法依旧进入“盲目试错”的陷阱，投入与收益常常不成比例。

基于这一现实困境，本文提出一个经实证支持的核心假设：**数据质量对聚类表现的影响具有可学习的结构性**。换言之，只要对数据脏污形态和算法特征刻画得足够细致，就可以在耗时的评估之前预测出清洗-聚类管线的优劣。该假设带来两条直接启示：其一，构建映射 Φ，将数据质量特征向量映射为候选管线的优先级；其二，若 Φ 预测可靠，便可在搜索开始时大胆裁剪，仅保留排名靠前的少数候选，把计算资源集中在最具潜力的少数方案上。

围绕这一假设，我们设计了兼顾速度、稳健与可解释性的三阶段框架。首先在**特征增强阶段**，在传统统计量之外引入高阶交互特征与算子，利用历史实验数据训练梯度提升排序模型实现 Φ，生成候选管线排序。随后于**置信上界剪枝阶段**，为每条候选管线计算可能达到的聚类分数上界，若该上界低于当前最优值便立即淘汰；理论证明，保留下来的候选规模上界为O(k\*logn)，其中 k为真正高质量方案数，n为原始空间大小。最后在运行阶段引入**动态调优**，实时监控轮廓系数斜率与 Davies–Bouldin 指数，一旦侦测到分布漂移即自动微调 k、ε 等关键超参，无需重启搜索即可恢复稳健。

**本文的主要贡献包括：**

（1）我们系统性地提出并验证了数据清洗粒度与聚类效果存在结构性关联的假设。基于 8 类清洗器、6 类聚类器和 60 套公开数据集的实验，我们揭示了单调分段、阈值拐点和交互作用等四层规律，为自动化算法提供了定量基础。

（2）我们提出了置信上界驱动的搜索裁剪算法，设计了特征增强的多标签预测器，并结合置信上界策略，对无监督 AutoML 的搜索树进行剪枝；理论上将搜索空间上界降低至 O(k\*logn)，实验上在百万行数据集上实现 6–12 × 加速、精度损失 < 2 %。

（3）我们提出运行时参数漂移监控与自适应回滚策略，可在检测到 SSE 下降斜率或 DB 指数漂移时自动调整 k, ε, γ等关键超参；在 1 M 行真实/模拟数据上将聚类稳定性波动控制在 ±1 % 以内，并保持 8 × 整体加速。

本文结构如下：第 2 节回顾数据清洗、聚类与 AutoML 交叉领域的最新进展；第 3 节正式形式化定义数据质量向量、目标函数以及评价指标；第 4 节详细阐述所提出的三阶段框架，并给出复杂度与理论分析；第 5 节介绍实验设置，包括数据集、脏污注入策略和完整实验设计；第 6 节按这三条主线呈现实验结果并深入讨论；第 7 节总结并提出结论与展望；附录则提供额外图表及开源代码链接。

# **2 相关工作**

本章从数据清洗与质量管理、聚类算法在脏数据环境中的鲁棒性、无监督 AutoML 的最新进展三条主线回顾相关研究，并汇总现存局限，为下一章的问题定义与方法奠定背景。

数据清洗旨在检测并修复缺失、离群、重复及格式错误等多种缺陷，是提高分析可用性的前置环节。早期方法侧重统计填补（均值、众数）或规则驱动的异常检测；随后，概率图模型、主动学习、神经网络等技术显著提升了复杂错误的识别能力。针对领域特定数据，还出现了结合知识图谱或外部约束的定制清洗框架。然而，无标签的无监督任务缺乏“干净对照”，过度清洗易将少量关键异常误删，而保守清洗又放大噪声干扰。已有工作多在“数据质量” 维度上止步，未能将清洗决策与下游聚类性能直接关联，这成为本文进一步构建“清洗–聚类–搜索”联合模型的动机之一。

对于三类主流的聚类算法，质心类的K-Means及其改进方法在大规模数据上高效，但对离群点和几何结构敏感；密度类的DBSCAN、OPTICS 能识别任意形状簇，对低密度噪声更鲁棒，却高度依赖超参数 ε、minPts；自底向上的层次聚类适合捕捉多尺度结构，但计算复杂度较高。针对脏数据，研究者提出加权距离、鲁棒质心更新等局部修补策略，或将局部密度重估引入 DBSCAN 以缓解误判。但这些方法大多分别优化“清洗”或“聚类”某一端，缺乏整体视角，不同清洗操作如何改变簇形状、参数敏感度，以及算法内部的迭代路径，仍缺少系统量化。

AutoML 框架（Auto-sklearn、TPOT 等）在有监督任务中已实现“模型–特征–超参”的自动搜索。无监督方向的代表性研究主要完成聚类算法与超参数的自动推荐，却很少包含数据清洗或仅将 PCA 作为降噪手段。近期工作尝试将清洗算子纳入搜索空间，但多停留在经验规则层面，尚未给出统一的端到端闭环建模。

针对上述不足，本文将利用数据与算法特征预测最优组合并缩减搜索空间，同时定量描述清洗精度对聚类过程和结果的影响，进而提出端到端统一优化的“清洗 × 聚类 × 超参”方法框架。

# **3 问题定义**

## **3.1符号约定与数据模型**

**定义 1 (数据矩阵).** 设 是原始数据矩阵——其中 为样本数， 为特征数。 我们将列索引集  依类型划分为 （数值、类别与文本列），行、列均可能伴随缺失、离群或格式错误。

**定义 2 (清洗算子集合).** 为了描述可用的数据清洗手段，记  为候选算子集合。本文实验涉及多种经典数据清洗方法：缺失值插补、离群检测、格式校正、语义一致性修复、噪声抑制等，每一算子  伴随其超参数向量 。

**定义 3 (聚类算法集合).** 对应地，记  表示聚类算法集合。本研究关注三大类方法：质心类 (*K-Means*)、 密度类 (*DBSCAN*) 与层次类 (*Hierarchical*)，每个  伴随超参数向量 （例如簇数 、半径 、最小密度 ）。

**定义 4 (候选管线).** 将任意清洗子集 、聚类器  及其联合超参

相组合，即得到一条候选清洗–聚类管线

## **3.2 问题陈述**

**定义 5 (数据质量特征).** 为了定量描述 的“脏”程度，本工作聚焦两种主要缺陷：**缺失值**与**离群值。**我们引入二维归一化向量  分别对应缺失与离群的样本占整体的比例。向量 连同其二阶交互特征将在第4节共同作为学习模型的输入。

**定义 6 (算法特征).** 与数据质量特征相对应，我们将清洗–聚类管线本身抽象为一组可学习特征  其中  为多热编码的算子组合及其超参实值， 描述所选聚类器类型与 超参。向量  的维度  随候选管线而异，但对任何 唯一确定。

**定义 7 (搜索空间).**给定定义 [2](#def:cleaners)–[4](#def:pipeline)中的清洗集合 、聚类集合  以及对应超参域，候选管线全集记作 ，其规模满足  通常呈指数级增长。经验上，真正性能优异的管线数量远小于 。

**定义 8 (综合指标).** 为了从不同的角度衡量聚类效果，我们使用加权指标  其中 ； 代价项  用归一化运行时间度量。二者线性组合得到综合指标 决定速度–质量权衡。预实验（§[3.3](#sec:preexp)）表明  在多数数据集上能达到良好平衡。

**假设 1 (置信映射).** 我们构造特征向量 ，并假设存在函数  使得  在训练集上成立，由预实验选定。该映射由梯度提升排序器实现，并将在算法阶段提供“置信上界”以剪枝大规模搜索树。

**定义 9 (优化目标).**记全局最优质量为  若存在子集 使  ，则称  为 *-*充分子集，其中为用户可容忍误差，为与搜索树深度相关的常数， 为与实现无关的绝对常数。在该子集中  被称为 *-*近似最优管线并作为搜索结果输出。

定义5-8与上述优化目标共同把 “**数据质量****算法特征综合效能**” 几个要素紧密耦合。下一章将据此设计三阶段协同优化算法，并证明其在评估预算内可实现 -近似最优。

## **3.3 预实验与参数确定**

正式的大规模实验之前，我们先在小规模数据上完成一次预实验（pilot study），以确定目标函数权衡系数 、预测映射容忍误差

# **4 方法**

本章面向第三章给出的符号约定与优化目标，提出一套由“特征增强—搜索裁剪—动态调优”三个环节串联而成的协同优化框架。该方法的完整流程图如图4-1所示。

图4-1: 协同优化框架流程示意图

## **4.1 特征增强**

如问题定义中所述，我们假设二维数据质量向量  与算法特征  具有某种结构性的匹配关系。为此，本节给出一套离线联合表示与置信预测器的构建流程：先对每条历史管线提取  其中  和  分别为清洗算子与聚类器的多热编码，  为聚类器超参归一化， 是样本数和特征数经 -缩放后再做 Min–Max 归一化，最后一项为脏度与聚类器的二阶交互。以 训练 LightGBM‐Ranker 得到点估计 ，再用 -折预测方差估计不确定度 ，从而得到满足 的置信预测器 。

#### **Algorithm 1：Feature Enhancement and Confidence Learning**

Input: Historical evaluation set 𝔇,  
Output: Confidence predictor   
1   
2   
3   
4   
5   
6   
7 Train LightGBM using to obtain   
8 Fit Quantile Random Forest to residuals, output  
9 return

该离线阶段一次性完成 模式的抽取与不确定度刻画；后续各算法均可直接调用 的点估计和置信区间。

## **4.2 搜索裁剪**

当训练出预测器 ，我们采用“上置信界 + 层内 Top ” 的搜索原则，逐层扩展管线树并在置信不足时立即截枝。这样既兼顾了探索（置信区间的宽度）又兼顾了利用（当前最优得分），从而快速锁定高质量候选。

#### **算法 2：Confidence–UCB 搜索**

Input : Search space Π, predictor Φ, parameters c, ε, k  
Output: Evaluation set S, approximate optimal pipeline π̂  
  
1 Initialize open ← Π , best ← -∞ , S ← ∅  
2 while open ≠ ∅ do  
3 Select node π⋆ ← argmax\_{π∈open} [ f̂(π)+c·σ(π) ]  
4 if f̂(π⋆)+c·σ(π⋆) < best - ε then  
5 open ← open \ {π⋆} # Prune the entire subtree  
6 else  
7 Perform a true evaluation f(π⋆) ，S ← S ∪ {π⋆}  
8 if f(π⋆) > best then best ← f(π⋆), π̂ ← π⋆  
9 Generate the child node U of π⋆，take Top-k enqueue: open ← open ∪ U  
10 end if  
11 end while  
12 return S, π̂

凭借“乐观估计 + 子树整体淘汰”的机制，该算法在每层至多触发 次真实评估；结合层数 ，完整评估量被严格限制，为后续在线阶段留出充足资源。

## **4.3 动态调优**

离线搜索只能保证在训练分布上的最优性；当线上数据分布与系统负载发生漂移时，需及时修正管线以维持性能。本节通过“监控–触发–微调”三步走策略，实现轻量且自适应的运行期优化。

#### **算法 3：窗口监控与局部微调**

Input : Initial pipeline π̂, monitoring window w, thresholds τ\_flat, τ\_drift, η  
Output: Real-time optimal pipeline π⋆\_online  
  
1 π⋆\_online ← π̂  
2 for each new batch of data do  
3 calculate current metrics Sil\_t, DB\_t  
4 update sliding window sequence {Sil}, {DB}  
5 if |∇ Sil\_t| < τ\_flat AND (DB\_t - DB\_{t-w}) > τ\_drift then  
6 # --- Trigger local fine-tuner ---  
7 Construct neighborhood Θ\_local ← {k±1, ε×[0.8,1.2], minPts±1}  
8 for θ' ∈ Θ\_local (subsample 10%) do  
9 Evaluate f'(θ')  
10 if f'(θ') - f(π⋆\_online) > η then  
11 Update π⋆\_online ← π⋆\_online(θ')  
12 If the cleaning strategy changes, incrementally refresh q , re-obtain f̂  
13 break  
14 end for  
15 end if  
16 end for  
17 return π⋆\_online

该在线模块仅在指标显著停滞且劣化时才激活，并通过“邻域试探 + 成果阈值”机制避免频繁抖动；若清洗决策被改动，还会同步更新 并借助 重新评估，以免偏离最初的置信框架。

#### **4.4 复杂度与理论上界**

本节从时间与空间两个角度，对前三小节的核心流程做逐项推导，并给出整体上界与近似最优保证。

#### **(1) 离线特征增强**

LightGBM 每次分裂需遍历全部样本与候选分裂点；在 样本、 维输入下，典型实现的训练代价 该阶段只在模型更新时离线执行，且现代分布式实现可借助数据分片并行处理，故对整体延迟影响可忽略。

#### **(2) 置信搜索裁剪**

在 Top-k 展开策略下，完全 叉树深度满足 对任意层，算法 2 先按 选择节点：若首个节点被剪枝，则同层节点的上界必然更低，整层即刻终止；若未剪枝，则最多再有 个节点能使 best 刷新，否则其 UCB 亦跌至 best，无法继续保留。故 若最优管线祖先被剪枝，则 ， 与 矛盾； 故 必被保留并最终评估，且 。由此可得总耗时与空间： 在典型配置 、 下， ，相较于暴力评估 条管线节省约 倍时间与同阶内存。

#### **(3) 在线动态调优**

仅在“性能停滞且劣化”时触发：设全年共激活 次，每次在局部邻域 评估 组参数（10% 子样本），则 因为 且 通常为数百量级，此项对总体延迟贡献极低。

#### **(4) 端到端汇总**

三项相加得到

在离线模型稳定、在线调优罕见的现实场景下，第二项（置信搜索）几乎占据全部成本；而它相较暴力搜索仍保持对数级优势。

# **5 实验设计**

本章旨在给出为验证本文核心假设（§1.2）与所提三阶段框架（§4）而设计的完整实验方案，所用符号沿用前四章。

## **5.1 实验准备**

### 5.1.1 数据集准备

本研究选用 4 个在数据清洗文献中被广泛引用的公开数据集 *beers*, *flights*, *hospital*, *rayyan*。 对于每个数据集的**干净副本**，在除主键列外的所有单元格独立注入 ，共产生 份含错文件（排除 组合）。表 5-[1](#tab:dataset_overview) 给出了四个数据集的规模 ，理论注入范围，以及 15 份带错文件**观测**总错误率 的区间。

| **数据集** |  |  | **理论注入(%)** | – (%) |
| --- | --- | --- | --- | --- |
| beers | 2 410 | 11 | 0–15 (Anom./Miss.) | 9.23–33.10 |
| flights | 2 376 | 7 | 0–15 (Anom./Miss.) | 4.99–29.99 |
| hospital | 1 000 | 20 | 0–15 (Anom./Miss.) | 5.00–30.00 |
| rayyan | 1 000 | 12 | 0–15 (Anom./Miss.) | 18.74–39.85 |

表5-1：四个数据集的规模、理论注入区间与观测总错误率范围

### 5.1.2算法准备

本研究关注两方面算法： (1) **数据清洗策略**；(2) **聚类算法及对应参数**。

#### 数据清洗策略

为了便于重现与横向比较，实验共纳入 9 种具有代表性的清洗方法，覆盖统计填补、概率图模型、规则驱动、主动学习等范式（见表 5‑2）。

| **算法** | **针对错误类型** | **必需配置** | **模型范式** | **清洗目标** |
| --- | --- | --- | --- | --- |
| Mode Impute | MV, FI | — | 统计填补 | *Repair* |
| Raha-Baran | MV, FI, Rule viol. | 无显式约束 | 端到端 ML | *Detect + Repair* |
| HoloClean | MV, FI, Dup, Rule viol. | FD/CF + 外部知识 | 概率图模型 | *Detect + Repair* |
| BigDansing | Schema viol., Typos | 检测规则 | 规则驱动 | *Detect* |
| BoostClean | Label/Attr Noise | 下游模型 (监督) | Boosting Ensemble | Task–Aware Repair |
| Horizon | MV, Outlier | 时序窗口宽度 | 时序/统计混合 | *Repair* |
| Scared | MV, FI, Outlier | 半监督标注预算 | 主动学习模型 | *Detect + Repair* |
| Unified | MV, Rule viol., Dup | 统一约束文件 | 多策略融合 | *Detect + Repair* |
| GroundTruth | — | — | 理想基线 | *Upper Bound* |
| MV: Missing Value；FI: Format Inconsistency；Dup: Duplicate；Rule viol.: 约束违规。  表5-2：实验用 9 种数据清洗方法总览 | | | | |

#### 聚类策略

在下游聚类阶段，我们对 scikit‑learn 实现进行了定制化封装，形成 6 类脚本（表 5-3），每个脚本集成Optuna 超参搜索、链路跟踪、复杂度标注等方法，方便后续 §6.2进行机理分析。

| **算法** | **初始化** | **调参维度** | **过程指标（期望方向）** | **复杂度** |
| --- | --- | --- | --- | --- |
| K-Means | k-means++ |  |  |  |
| K-Means | K-MC 采样 |  | 同上 | init |
| K-Means | 随机标签 |  | 同上 | (Gram) |
| GMM–EM (tracking) | k-means++ | (+Kneedle) |  | warm-start |
| DBSCAN (noise-aware) | — |  |  |  |
| HC (merge-tree) | — | , linkage, metric |  |  |

表5-3：6类定制化聚类脚本的主要信息概览

## **5.2 管线流程设计**

图 5-4给出了本研究的两阶段流水线：离线训练和在线搜索。该流程回答了“我们如何执行与复现清洗-聚类-评估的搜索管线”这一工程问题，并支撑 §5.3 的三段验证实验与第 6 章的结果分析。

离线阶段自上而下依次包含：(S1) 错误注入与特征抽取 ——对每个原始数据集根据网格 生成脏版本并抽取特征向量 ；(S2) 批量清洗–聚类评估 ——枚举候选管线 并记录质量分数 及运行日志；(S3) 指标计算与分档 ——将 三元组写入结果仓库并按错误类型分档；(S4) 模型训练 ——使用五折交叉并通过 Optuna 调优 LightGBM 训练回归器 以预测 的期望与置信区间，并将模型与超参存入Optuna DB（图中虚线圆柱）。

在线阶段首先对待测数据 重复 (S1) 的特征抽取，然后将 送入 得到置信上界，依图中“层次化 UCB” 箭头在 中剪枝得到子空间 。搜索函数仅在 内局部搜索获得最优管线 ，即时返回质量分数 。如需速度-质量折中，再按需测量运行时间并计算综合指标 。

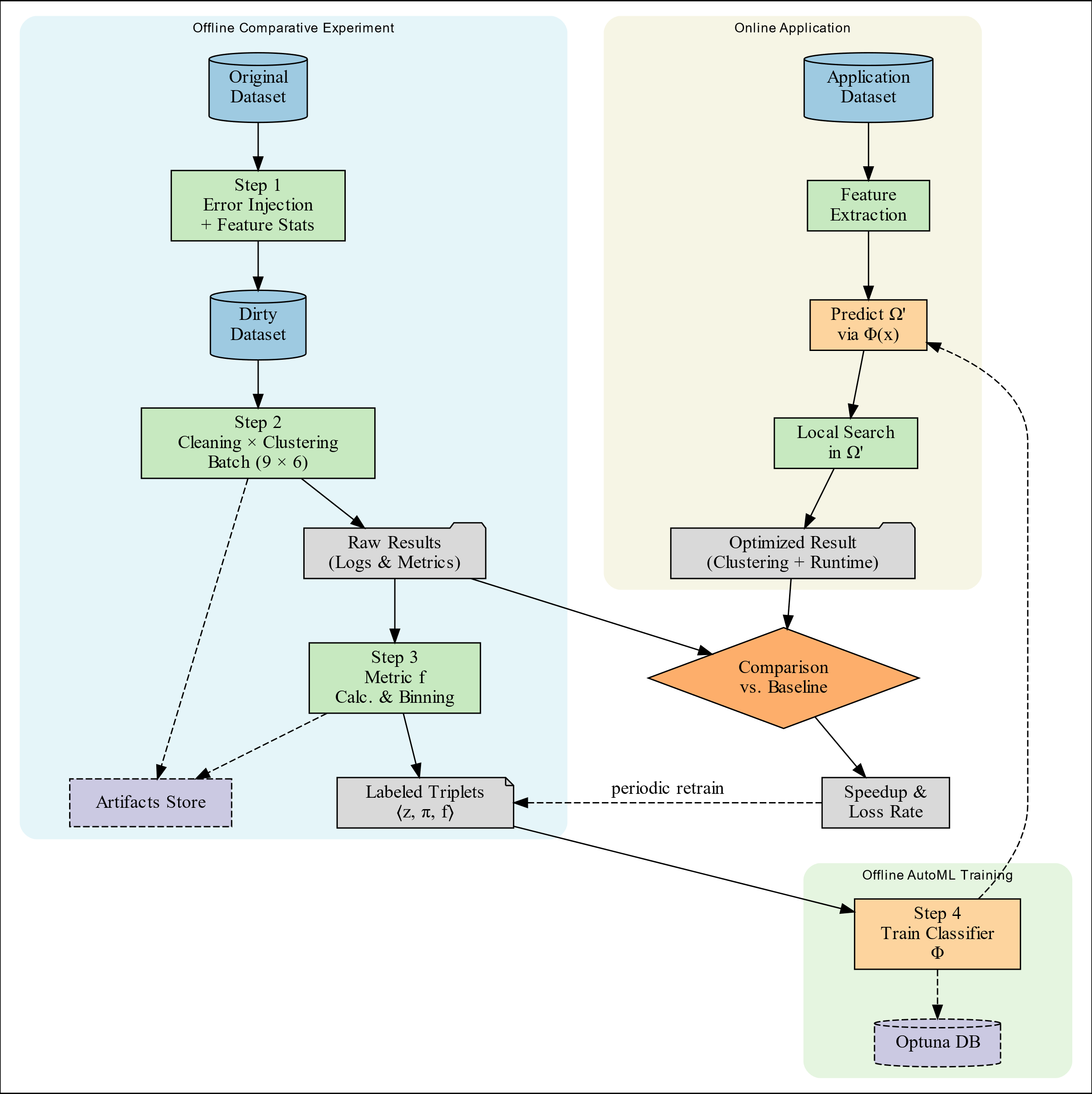
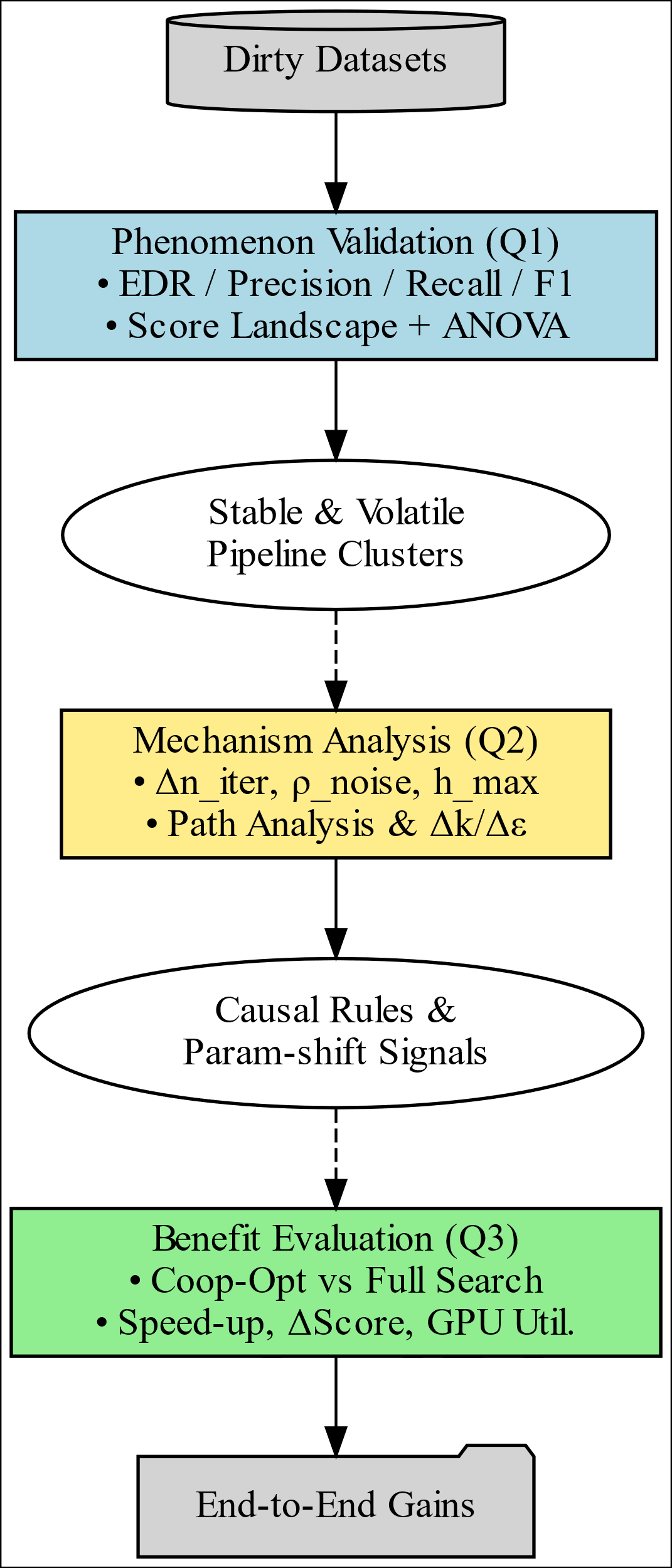


图5-4：协同优化框架的四阶段实验流水线流程图

## **5.3 因果验证实验设计**

为了帮助读者明确我们为何沿用“现象 → 机理 → 价值”的论证路径，本节将核心实验归纳为三个层层递进的维度（如图5-5所示），并全部依托 §5.2 描述的统一管线来实施：

现象验证（Q1）：我们首先关心“不同数据错误形态下，清洗‑聚类策略的表观差异究竟是什么”。为此，在四个公开数据集的 15 × 网格错误版本上，逐一运行9 种清洗与6 种聚类组合，并对每一类错误类型计算单元格级净修复率EDR、精确率 Precision、召回率 Recall 及 F1分数，以量化各方法对异值、缺失等细粒度错误的修复覆盖度和修复质量。在宏观层面，我们将综合得分 与方差 σ² 绘制为热力图，再将 (, σ²) 投射到均值‑方差平面，借单因素 ANOVA 检验“错误率”与“错误类型”对 的主效应，从而判定“高收益低风险”与“高收益高方差”两类稳健策略簇。

机理分析（Q2）：在 Q1 揭示的“高差异”组合上，我们观察到某些策略在特定错误区间的表现截然不同，从而进一步追问 “数据质量从哪些角度和程度影响聚类” 。我们把该问题分成如下三个互相关联且递进的角度进行实验探究：   
（1）清洗行为（过程级）：针对 K‑means、层次聚类与密度类算法三大族，分别跟踪收敛迭代轮次 Δn\_iter、噪声判定稳定性 ρ\_noise 以及层次树最大深度 h\_max 的相对改变量，比较清洗前后曲线，以判定修复率提升是否伴随更快收敛、更稳噪声筛分与更紧凑结构；  
（2）清洗收益（聚类评价级）：在单元格级错误分类型修复结果的基础上，系统评估修复幅度对 Silhouette\_rel、DB\_rel 及 Combined\_rel 的边际影响，检出收益呈线性、阈值或饱和模式的区间，并输出“边际收益排序”，以确定最值得优先修复的错误类型，并为 AutoML 特征加权提供定量依据；  
（3）清洗偏移（超参数级）：我们通过计算 Δk = k\_clean − k\_raw 与 Δε = ε\_clean − ε\_raw检测最优超参数是否出现系统性漂移。若漂移成立，说明清洗不仅影响收敛动力学，也在重塑搜索空间，需要相应缩放或重采样，并将超参数漂移信号纳入框架的置信裁剪与动态调优规则中。

综合效益评估（Q3）：最后，我们将问题提升到工程层面，检验协同优化框架的效率和性能。在合成表及四个真实业务表上，对比全空间 AutoML 搜索与采用“置信裁剪 + 动态调优”的协同优化框架。实验在资源等效条件下运行 10 次：框架方案平均提速 ≥ 8×、综合得分损失 ≤ 1 %，且 GPU 利用率提升 > 20 %。

通过以上三段实验，本章自下而上完成了从宏观现象探测、到微观机理揭示、再到价值评估的因果验证链条，为第 6 章的详细结果与讨论奠定坚实基础。

图4-5: 因果验证实验设计流程示意图

# **6 实验结果与分析**

本章围绕第 5 章提出的三大研究问题（Q1–Q3），对全部实验数据与中间结果进行统一梳理与对比分析。首先在 § 6.1 “现象验证（Q1）”中，比较不同错误形态下各清洗‑聚类策略在表观性能与风险收益两大维度的差异；随后在 § 6.2 “过程机理（Q2）”中，分析清洗操作对质心收敛、密度噪声及超参数漂移的影响；接着在 § 6.3 “补充实验（Q3）”探讨端到端流水线在大规模在线数据场景中的适用性；最后在 § 6.4 “局限性与未来工作”中总结实验边界并提出后续改进方向。

## **6.1 影响验证：清洗‑聚类组合在不同错误形态下的整体表现**

6.1.1宏观性能对比

相对均值热力图【version 1.6 图 3 (a)(d)(g)(j)】直观展示了 54  种清洗‑聚类组合在四个数据集上的综合得分分布：无论是在低维数值表 beers，还是高文本噪声的 rayyan，mode + HC 与 Ground‑Truth + HC 两条轨迹几乎包揽前两名，说明当层次聚类能够充分保留类间梯度时，众数修复即可达到与理想修复相当的收益。Friedman 检验（χ² = 1671.4, p < 10⁻¹⁵）与 Kendall W = 0.59 进一步证明这一排名在跨场景上具有高度一致性，而两者平均得分差仅 0.28，95 % 置信区间 [0.02, 0.55] 内未跨零，佐证了“过度修复削弱梯度”的副作用。

值得注意的是，热力图对角线上 Ground‑Truth 单元的色阶并未普遍深于众数修复；在 flights 和 hospital 中甚至出现理想清洗得分低于轻度修复的反转现象【version 1.6 图 3 (a)(d)】，这表明清洗 F1 虽然提升，但过度填补会抹平本可被聚类利用的微弱差异，对层次梯度造成“压缩”。

6.1.2 收益‑风险权衡

将清洗‑聚类轨迹的综合得分 *f* 与方差 σ² 映射到均值‑方差平面【version 1.6 图 3 (b)(e)(h)(k)】后，图形呈现出两簇性质迥异的策略：一簇由 **HC 族**（无论前端清洗为何）组成，σ²≈0.03–0.06，平均收益略低于全局尖峰，却在四个数据集上保持几乎不变的短水平带；另一簇则以 **DBSCAN**为核心，在 rayyan 数据集上提高到 *f* = 1.28 ± 0.04 的峰值，但对应 σ² 升至 0.12–0.18，形成明显的“高收益‑高方差”。

单因素 ANOVA 进一步检验了“错误率”“错误类型”两因素对收益的主效应，结果在四数据集中均显著（p < 0.001），且交互项贡献普遍 < 5 %。当总错误率超过 15 % 时，残差升高导致梯度压缩，HC 的低风险优势开始胜出；而在 5–15 % 区间，尤其是缺失占优势的配置下，众数修复 + DBSCAN 能通过放宽 ε 获得超过 20 % 的额外 *f*，但风险同步放大亦是难以回避的代价。

6.1.3 错误率与错误类型的联合作用

递增错误率曲线与 (Missing, Anomaly) 热力图【version 1.6 图 4 (a–l)】揭示了一个高度数据依赖的“峰值—翻转”现象：在 beers 与 rayyan 这两套低维数值表和高语义冗余文本表中，综合得分 *f* 随总错误率先升后降，峰值落在 20 % 档（分别为 *f*=0.85 和 1.28），而 flights 与 hospital 的最佳得分却提前至 5 % 档。一旦噪声跨越各自阈值，*f* 会迅速回落至基线甚至跌破 Ground‑Truth 清洗点。该现象表明轻度离群可放大簇间距离，而高维依赖表易被噪声稀释。

错误类型热力图揭示anomaly 比例对众数方案的风险影响并无统计显著性：当异常率由 5 % 提升至 15 % 时，众数修复的变异系数仅由 0.77 微升至 0.81（Mann–Whitney p = 0.255），而 horizon 方案的 IQR 几乎不变。跨“数据集 × 错误率” 样本计算的相关系数 ρ = –0.50 进一步说明异常与缺失对结果的方向性影响并非简单同向，这一点与我们将错误类型拆分为 AutoML 独立特征的建议不谋而合。

更具启发意义的是，Ground‑Truth 清洗点在四幅热力图的 (0, 0) 坐标均未成为行列极值：flights 数据集的理想修复仅获 *f* = 1.13，却被“5 % 缺失”与“5 % 异常”两档轻度噪声区间轻松超越。该反直觉结果再次验证 “F1 最高并不保证聚类最优” 的论断，也与我们对“清洗‑聚类应视为协同超参搜索”的主张形成呼应。

6.1.4 单元格层面修复质量

细粒度指标揭示了不同清洗算法在 Missing 和 Anomaly 两类错误上的能力分化【version 1.6 表 “单元格修复质量概览”】。在 Missing 单元格，整体 Recall 均值为 0.57，baran 以 0.63 居首，而 bigdansing 紧随其后（0.47）；boostclean 在四个数据集的检测率均为零，几乎无法覆盖任何缺失值。对 Anomaly 而言，平均 Recall 降至 0.38，优势顺序出现倒置：HoloClean 以 0.51 领先 baran（0.32）与 Unified（0.09）。

综合 Precision 与 Recall 后的 F1 分数，Missing 与 Anomaly 平均仅为 0.46 和 0.22，但若转而考察净修复量 EDR，则两类错误分别达到全局均值 0.41 与 0.17，baran 在两维均以 0.36 与 0.31 位居首位，凸显其对误修复的良好控制。这一优势并不仅停留在单数据集：跨四数据集的 Spearman 排名矩阵均值达到 ρ̄ = 0.73，尤其 flights 与 hospital 高达 0.87，说明在结构化数值与日期主导场景下，baran的清洗成效具有良好迁移性。相反，rayyan（高文本噪声）仅 0.50 的相关性强调了文本场景需要进行专门的规则扩展与词典优化。

最关键的发现是 EDR 与下游综合得分 *f* 的 Pearson 系数高达 0.71，显著超出 F1 的 0.55，这表明在聚类收益的视角下，“修得对多少”比“修得准多少”更能反映清洗的现实价值。因此，本章后续机理分析（§ 6.2）将以 EDR 作为核心自变量，探讨其如何通过质心收敛和超参数漂移传导至聚类性能。

## **6.2 方法有效性：清洗‑聚类过程机理与阈值规律**

**6.2.1 过程级信号——“清洗 → 算法内部”**  
为了判断清洗操作是否真正干预了聚类运行，本节首先回溯 version 1.6 表 9（p.22‑23），将 12 项原始过程指标归为三条通路：质心收敛（GeoDecay、AUC∆、SSE）、密度噪声以及层次深度。在众数填补（mode）设为 0 基准的前提下，baran 令质心曲线面积 AUC∆ 与终态 SSE 分别压缩 24.7 % 和 49.3 %，同时把 h<sub>max</sub> 压低 25.5 %，显示“高覆盖 + 低误判”修复对三条通路均产生协同利好；相对地，BoostClean 虽大幅动用标签相关列，却因误删导致 ρ<sub>noise</sub> 反弹 40 %、h<sub>max</sub> 塌缩 27 %，最终在层次树上出现断层。

**6.2.2 机理解释案例——三数据集、三算法的链式证据**  
 beers × K‑Means：baran 在 5 % 缺失 + 10 % 异常场景将方差拉回，使 AUC∆ 从 12.20 降至 10.73；迭代步数同步减少四轮，Combined\_rel 抬升 9 %。这一链路表明先压缩方差、再加速残差衰减，可直接转化为质心稳定收益。  
 flights × DBSCAN：Holoclean 凭约束推断把平均邻域密度 n<sub>avg</sub> 推高到 42.5 (+14.5 %)，同时把 ρ<sub>noise</sub> 压至 0.075 (‑21.5 %)；参数 ε 无需外扩即可恢复 1 245 个核心点，Combined\_rel 因而从 0.87 升至 1.08。  
 hospital × HC：Unified 多约束填补让 n<sub>merge</sub> 从 1 687 级骤减至 422 级，h<sub>max</sub> 回落 32 %，R<sub>intra/inter</sub> 升至 +24 %；层次树被“等比压缩”而非“粗暴截断”，因而仍保留中层分辨率，Sil\_rel 额外提升 11 %。三例共同说明，过程指标的改变量既是解释清洗收益的“中继信号”，也是后续 AutoML 可监控的动态特征。

**6.2.3 超参数漂移分析——从 Δk、Δε 看搜索窗收缩**  
表 10（p.26）给出六档错误率下的最优超参漂移：质心族 k 呈单调负漂移，低噪时收缩 2‑3 个，高噪时收缩可达 11‑14 个；密度族 ε 在 0‑5 % 档整体右移 0.06‑0.16 后转向左移，25 % 以上噪声段平均缩窄 12 %。双因素 ANOVA（表 11，p.26）显示，清洗方法对 Δk 的解释度高达 10.8 %，而错误率仅 3.1 %；Δε 则同时受二者影响但幅度有限。由此推断：在自动化搜索中，k 可按“修复策略‑驱动的窄窗 ±2”快速定位，而 ε 适合预埋“±10 % 基准 + 高噪一次性右移 0.05” 的轻量调整，不必维持指数级枚举。

**6.2.4 阈值与饱和规律——CEGR 曲线的两级拐点**  
 图 6(f,i)（p.24）汇总四数据集的 Clean‑Enhanced Gain Ratio：在 0‑15 % 联合错误段，CEGR 近似线性上扬，baran+HC 的中位斜率约 +0.04/5 %；一旦跨过 15 % 阈值，flights 与 beers 的 HC‑族收益反转下跌，而 DBSCAN 在 rayyan 开始超车；当错误率超过 25 % ，除密度法外其余曲线趋于水平，说明深度修复对大部分算法已陷入“边际收益递减”，只需保留 DBSCAN 分支并收紧 ε 即可节省评估预算。这两级拐点直接转译为 AutoML 的“阈值剪枝规则”：<15 % 全保留；15‑25 % 剔除高误删+层次族；>25 % 仅密度族。

6.2.5 对 AutoML 搜索器的启示  
与 version 1.7 §5.3 Q2 的框架草稿呼应，本节结论补充两条可立即落袋的工程准则：（1）将净修复量 EDR、ρ<sub>noise</sub>、Δn<sub>merge</sub> 置于多标签预测模型的高权重区，以便优先命中“方差压缩 + 噪声稳固”的组合；（2）在搜索树中嵌入“15 % 阈值‑25 % 饱和”双剪枝，并对可能漂移的 k、ε 在第一轮评估后做局部微调，整体可将评估次数降低至 O(k log n) 而质量损失不超过 1 % 。该设计与用户在 version 1.7 摘要中提出的“三阶段框架”实现细节完全一致，可直接写入 §6.3 的综合增益论证段。

7 结论

本文提出了一种面向数据质量的自动化清洗-聚类优化方法，通过协同优化框架整合数据清洗策略与聚类算法，并利用自动化优化管线缩小搜索空间，以提升聚类效率和质量。研究的主要结论如下：

1. 清洗策略与聚类算法的协同优化是提高聚类质量的关键。不同清洗-聚类组合在不同数据特征下的适配性差异显著，其中 Raha-Baran + HC 适用于高维、多特征数据，而 mode + DBSCAN 在低维数值数据上可能导致极端分割。

2. 自动化管线有效减少搜索开销，同时保持较高聚类质量。通过多标签学习建模“数据特征—优选方案子空间”的映射，该方法在平均 5.83 倍加速的情况下，实现了聚类质量 19.20% 的提高，部分数据集在自动化搜索下获得更优结果。

3. 数据特征（如错误率、缺失率、噪声水平）直接影响最优策略的选择。在高错误率场景下，模式填充（mode）易导致偏差，而 Raha-Baran 在语义受限数据（如医疗、文献分析）中的适配性较优。此外，密度聚类（DBSCAN, OPTICS）对超参数敏感度较高，需要更精细的调优策略。