# 共享单车数据分析与强化学习调度——项目计划（v1.0，2025-10-26）

目标用一句话：**用 10 万条历史数据做分析和需求建模，用自建模拟器进行交互采样，训练并评估共享单车调度策略（RL + 启发式基线），形成可演示的平台与报告**。

## 1. 项目目标（Goals）

**分析平台**：基于 Spark/Pandas + Pyecharts/Flask 展示共享单车多维度分析（时段、季节、天气、用户/车辆）。

**需求建模**：利用 Kaggle 历史数据校准“小时 × 天气 × 季节 × 工作日”需求强度函数 λ(t)。

**调度模拟器**：搭建城市分区的 Gym 环境，模拟库存流动、需求到达与调拨成本。

**强化学习**：在模拟器中训练策略（PPO/DQN），并与启发式基线对比（服务率/缺口/成本/收益）。

**可视化与演示**：Dashboard 增设“策略对比”“What-if 场景”，支持一键重跑小规模模拟并展示结果。

### 成功标准（Acceptance Criteria）

可视化页面可展示：日/小时趋势、季节/天气影响、用户/单车 TopN；

模拟器可运行：支持≥5个区域、≥365×24 小时仿真、支持夜间/逐小时两种调度节奏；

训练至少 1 种 RL 策略（PPO 或 DQN），并与≥2种基线对比；

输出统一评估表与图：服务率、未满足需求、调度成本、总收益、鲁棒性（不同天气/季节/预算）。

## 2. 数据策略（Data Strategy）

**历史数据量级**：扩充到 **~10 万条订单**，用于分析展示与需求函数拟合；无需为 RL 训练造“百万级离线轨迹”。

**来源**：Kaggle Bike Sharing（hour/day 聚合）+ 自扩数据（订单明细、用户/车辆表）。

**用途分工**：

**分析/展示**：10万条订单明细；

**需求校准**：基于 hour.csv 字段，拟合 λ(t)（小时、季节、工作日、天气）；

**模拟采样**：RL 训练时按 λ(t) 在线采样，不落盘。

**数据质量**：

字段一致性：时间、时段、工作日、天气、区域；

合理分布：早晚高峰、周末/工作日差异、季节性；

稳健性检查：异常值、缺失值、分布对齐（KS/QQ）。

## 3. 需求模型（Demand Modeling）

**形式**（例）：log λ\_t = β0 + β\_hour[h] + β\_season[s] + β\_weekday[w] + β\_weather[k]；

**估计方法**：

基线：Poisson/Quasi-Poisson 回归；

可选：Gradient Boosting（XGBoost/LGBM）对 λ\_t 回归，再做截断与平滑；

**空间拆分**：城市划分为 K 个区域（K=5~12），给定权重 w\_z，λ\_{z,t} = w\_z \* λ\_t；权重可由：热点热力、POI 经验或等分设定。

**校准与验证**：

回测：用 2011/2012 的天/小时曲线与模型拟合对齐；

稳健性：在极端天气/节假日场景下 λ(t) 合理收缩/放大；

指标：RMSE/MAE、高峰时段捕捉率、周末/工作日差分趋势一致性。

## 4. 调度模拟器（Gym 环境）

**状态（State）**

B\_z：各区可用车辆库存；t：时间索引（0..T-1）；

上下文：hour(t) / weekday(t) / season(t) / weather(t)；

可选：调度车容量、在途车辆、区域容量上限、需求未满足队列等。

**动作（Action）**

**夜间集中调度**（简化版）：每日末时段一次性从富余区→短缺区调拨；

**逐小时滚动调度**（进阶）：每步给出若干 (i→j, qty) 决策；

约束：总调拨量上限、单次最大流量、车辆/卡车容量与路径成本表。

**转移与需求**

每步按 D\_{z,t} ~ Poisson(λ\_{z,t}) 抽样；

满足量 S\_{z,t} = min(D\_{z,t}, B\_z)，未满足 U\_{z,t} = D\_{z,t} - S\_{z,t}；

库存更新：B\_z ← B\_z - S\_{z,t} + inbound - outbound；

动作在下一时刻/延迟若干步生效（可配置 lead time）。

**奖励（Reward）**

Revenue = p \* Σ\_z S\_{z,t}；

Penalty = α \* Σ\_z U\_{z,t}（未满足惩罚）；

RebalanceCost = Σ\_{(i→j)} c\_{ij} \* qty；

r\_t = Revenue - Penalty - RebalanceCost（可归一化/折扣）。

**情景与随机性**

天气/季节/工作日驱动 λ(t)；

罕见高峰（活动日）触发倍增系数；

参数可通过 UI 进行 What-if 配置。

**实现建议**

接口兼容 gymnasium：reset(seed), step(action), observation\_space, action\_space；

config.yaml：区域数K、时间跨度T、成本矩阵、预算、延迟、生效节奏等；

单元测试：库存守恒、成本计算、边界条件（全零/全满/极端需求）。

## 5. 策略与算法（Policies & Algorithms）

**启发式基线**（优先落地）

**Zero-Action**：不调度，仅记录服务率与缺口；

**Proportional Refill（比例补货）**：按历史均值占比分配库存；

**Min-Cost Flow（简化）**：按缺口与富余构图，贪心匹配最短路径补给；

**RL 算法**（二选一先跑通）

**PPO**（离散：多维动作模板；连续：对每条边给出 qty，再裁剪/归一化）；

**DQN/DoubleDQN**（离散动作：预定义调度模板/候选方案）。

**训练与评估**

Episode：T = 365×24（可先用 T=90×24 预热）；

指标：服务率（S/D）、缺口总量、RebalanceCost、总收益、稳定性（随机种子/情景切换）；

卫生度：训练曲线（reward/服务率）、随机种子均值±方差；

模型输出：策略快照（.zip）、评估报告（CSV/HTML 图）。

## 6. 可视化与平台集成（Dashboard）

**分析页**：日/小时趋势、季节/天气影响、TOP 用户/单车、车型/性别分布。

**仿真页**：

场景设置：区域数、预算、天气、是否工作日；

策略选择：Zero / Proportional / MinCost / PPO / DQN；

指标对比：服务率、缺口、成本、收益（折线/柱状）；

热力图：t 时刻的各区库存 vs 需求热度；

一键重跑：小规模 T（如 7×24）即时演示。

**实现**：Pyecharts 生成 HTML 片段 + Flask 模板渲染；/api 暴露一次仿真接口返回 JSON。

## 7. 里程碑与时间表（Milestones & Timeline）

当前日期：2025-10-26（周日）

**M1｜10-27 ~ 10-29** —— 数据与分析（3 天）

完成 10 万行订单扩充脚本与质量检查；

拟合 λ(t)（Poisson/GBDT），输出校准曲线；

完成分析页 1.0（趋势/分布/TopN）。

**M2｜10-30 ~ 11-01** —— 模拟器 1.0（3 天）

实现 Gym 环境（夜间调度版），通过单测；

接入 λ(t) 在线采样，跑通 Zero/Proportional 基线；

输出评估 CSV 与对比图表。

**M3｜11-02 ~ 11-04** —— RL 训练与对比（3 天）

接入 PPO（stable-baselines3），完成若干超参试验；

增加 Min-Cost Flow 简化基线；

产出对比报告与可视化。

**M4｜11-05 ~ 11-07** —— 平台集成与演示（3 天）

Flask 集成仿真页 + What-if 面板；

一键重跑 7×24 小时演示；

文档与 PPT 1.0，彩排。

机动缓冲：11-08 ~ 11-09（修 bug / 打磨图表与讲稿）

## 8. 任务分解（Work Breakdown）

**数据与建模**

**模拟器**

**RL 训练**

**可视化与平台**

**文档与交付**

## 9. 技术选型与实现细节

**环境**：Python 3.10+，PySpark 3.3.x（分析），Flask + Pyecharts（前端），stable-baselines3（RL）。

**代码结构**（建议）：

bike-sharing-rl/

├─ data/ # 历史与扩展数据

├─ demand\_model/ # λ(t) 回归与参数

├─ simulator/ # gym 环境与基线策略

├─ rl/ # 训练脚本与模型

├─ analysis/ # Spark/Pandas 分析

├─ web/ # Flask 与前端

└─ reports/ # 评估结果与图表

**性能**：本机 8~16GB 内存下，10 万行分析 < 5 分钟；仿真 7×24 小时实时演示秒级；训练使用较小环境先验（K≤8、T≤90×24）。

## 10. 风险与对策（Risks & Mitigations）

| **风险** | **影响** | **对策** |
| --- | --- | --- |
| λ(t) 拟合不足 | 仿真不可信 | 加入分层特征（月份/节假日），使用 GBDT 校正； |
| 动作空间过大 | RL 难以收敛 | 先做夜间集中调度 + 模板化动作； |
| 训练不稳定 | 指标波动 | 固定随机种子、归一化奖励、加熵正则； |
| 计算资源不足 | 训练耗时 | 降低 K 与 T；先用小环境调参； |
| 演示卡顿 | 观感差 | 预生成一个“最佳策略”结果作为兜底。 |

## 11. 交付物清单（Deliverables）

**分析页面**（HTML + 截图）；

**需求模型报告**（λ(t) 回归结果与曲线图）；

**模拟器代码**（含单元测试）；

**基线与 RL 策略**（训练脚本、参数、模型文件）；

**评估报告**（CSV/图表）：服务率/成本/收益对比；

**Flask 演示站点**（含仿真页与 What-if 面板）；

**PPT**（故事线、方法、结果与展望）。

## 12. 后续扩展（Roadmap+）

逐小时滚动调度 + 运输时延；

区域图嵌入真实路网（最短路/成本来自 OSM）；

需求预测从回归升级为时序模型（Prophet/Temporal Fusion Transformer）；

Offline RL（若未来有真实“动作日志”）；

Docker 化与一键部署脚本。

### 附：评估指标定义

**服务率**：Σ S / Σ D

**缺口量**：Σ U

**调度成本**：Σ c\_ij \* qty

**总收益**：Σ Revenue - Σ Cost

**稳健性**：跨天气/季节/预算的均值±方差

本计划作为 v1.0，可按里程碑推进并在每个阶段末更新版本。