Final Report: Dynamic Pricing for Urban Parking Lots

Summer Analytics 2025
Organized by Consulting & Analytics Club in collaboration with Pathway

---

1. Project Overview

This project aimed to design a dynamic, real-time pricing engine for urban parking lots that could:

- Dynamically update prices based on occupancy and demand
- Adapt to environmental and traffic factors
- Account for competition between parking lots

The models are developed with explainability and adaptability in mind, supporting intelligent routing and revenue optimization for city infrastructure.

---

2. Dataset Description

- **Time Frame:** 73 consecutive days
- **Locations:** 14 distinct parking lots
- **Time Intervals:** 18 per day (30-minute slots)

Key Features: - `Occupancy`, `Capacity`, `Queue`, `VehicleType` - `Traffic Conditions` (Low, Medium, High) - `SpecialDay` indicator (binary) - Geospatial Coordinates: `Latitude`, `Longitude`

---

3. Data Preprocessing

Steps performed: - Removal of rows with corrupted Latitude values - Conversion of Latitude/Longitude to float - Dropping missing values (NaNs) - Conversion to proper datatypes for numerical and spatial calculations

---

4. Modeling Approach

Model 1: Linear Occupancy-Based Pricing

A baseline model increasing price proportionally to occupancy.

Formula:

$$\text{Price}_{t+1} = \text{Price}_t + \alpha \times \left(\frac{\text{Occupancy}}{\text{Capacity}}\right)$$

Parameters: - Base Price = $10 - $\alpha = 2.0$

Model 2: Demand-Based Pricing

A weighted model that integrates various demand drivers.

Demand Score Formula:

$$\text{Score} = \alpha \times \text{OccRatio} + \beta \times \text{Queue} - \gamma \times \text{Traffic} + \delta \times \text{SpecialDay} + \varepsilon \times \text{VehicleTypeWeight}$$

Price Calculation:

$$\text{Price} = \text{Base} \times (1 + \lambda \times \text{NormalizedDemand})$$

Parameters: - Base Price = $10
- $\lambda = 1.0$ - Price bounded between $5 and $20

Model 3: Competition-Aware Pricing

Incorporates prices from nearby parking lots within a 0.5 km radius.

Steps: - Used `geopy` to calculate distances between lots - Computed average competitor price in local radius - Adjusted own price based on deviation

Formula:

$$\text{FinalPrice} = \text{OwnPrice} + \lambda_c \times (\text{OwnPrice} - \text{AvgCompetitorPrice})$$

Parameters: - $\lambda_c = 1.0$ - Final price bounded between $5 and $20

---

5. Outputs

Final output includes calculated prices for each timestamp and lot: - `LinearPrice` - `DemandPrice` - `CompetitivePrice`

Stored in: `final_pricing_output.csv`

---

6. Assumptions

- Uniform Base Price = $10 across lots
- Medium traffic assumed for 'average' values
- Weights based on domain knowledge and experimentation
- Competition modeled within a 0.5 km radius only

---

7. Submitted Files

- Jupyter Notebook (`.ipynb`)

- Final Report (`.pdf`)
- Output CSV (`final_pricing_output.csv`)

8. Conclusion

This project effectively demonstrates a scalable, modular pricing engine that can respond to real-world urban dynamics. The models combine simplicity, interpretability, and spatial awareness — providing a strong foundation for dynamic parking price management in smart cities.