Capstone Project Report

Dynamic Pricing for Urban Parking Lots

Capstone Project Report: Dynamic Pricing for Urban Parking Lots

■■■ Submitted by: Kaushal Jha

■ Program: Summer Analytics 2025

■ Host: Consulting & Analytics Club × Pathway

Project Overview

Urban parking spaces are limited, especially in high-demand areas. Static pricing often leads to:

- Overcrowding during peak hours
- Underutilization during off-peak hours

Objective:

Develop a real-time, intelligent pricing system for 14 parking lots using:

- Historical + real-time data
- Demand factors (occupancy, traffic, queue, vehicle type)
- Competitive pricing logic
- Only Python, NumPy, Pandas, and Pathway (no ML libraries)

Dataset Summary

- 14 parking lots
- 73 days × 18 time points/day (30 min intervals from 8:00 AM – 4:30 PM)
- Each record includes:
- Occupancy, capacity, queue length
- Latitude, longitude
- Vehicle type
- Nearby traffic conditions
- Special day flag

Models Implemented

Model 1: Linear Occupancy-Based Pricing

Formula:

$Price_{t+1} = Price_t + \alpha * (Occupancy / Capacity)$

- Base price starts at $10
- Price increases linearly as occupancy increases
- Ensures minimum price = $5 and maximum = $20

Purpose: Serve as a simple, interpretable baseline model

Model 2: Demand-Based Pricing

Demand Function:

$Demand = \alpha*(Occupancy/Capacity) + \beta*QueueLength - \gamma*TrafficLevel + \delta*SpecialDay + \varepsilon*VehicleTypeWeight$

Final Price:

$Price = BasePrice * (1 + \lambda * NormalizedDemand)$

Weights Used:

- Occupancy Ratio: 0.6
- Queue Length: 0.3
- Traffic (inverse): 0.4
- Special Day: 0.8
- Vehicle Type: 0.5
- Demand Sensitivity: 0.5

- Prices are bounded between $5 and $20
- Demand is normalized before applying

Model 3: Competitive Pricing (with Location Awareness)

Additions over Model 2:

- Calculate distance between parking lots using Haversine formula
- Find competitors within 1 km
- Adjust price based on nearby pricing:

If:

- Lot is full + nearby cheaper lots → Decrease price or reroute suggestion
- Nearby lots are expensive → Slightly increase price

Result: Smarter pricing strategy that accounts for customer alternatives

Real-Time Integration with Pathway

- Used Pathway to simulate data ingestion in timestamp order
- Streaming applied using .select(pricing_logic)
- Real-time Bokeh plot via .plot() on delta_window
- Panel used to serve the interactive visualization

Visualization

- Bokeh plots show real-time price updates per lot
- Hover tool reveals timestamp, occupancy, and final price
- Optional line comparison between your lot and nearby competitors

Assumptions

- Demand normalization assumes max demand value ≈ 10
- Traffic levels mapped: low=1, medium=2, high=3
- Vehicle weight mapping: bike=0.3, car=1.0, truck=1.5
- Nearby lots defined as those within 1 km radius

Insights

- Prices increase during late mornings and early afternoons
- Trucks typically trigger higher prices due to weight factor
- Special events lead to more aggressive price scaling
- Competitive pricing smoothens out extreme fluctuations

Deliverables

- ■ Colab notebook with all models
- ■ Real-time simulation via Pathway
- ■ Visualizations using Bokeh
- ■ Final report (this file)

Conclusion

This project shows how intelligent pricing systems can significantly optimize parking space utilization.

By combining simple logic with real-time data and competition signals, this pricing engine mimics a real-world revenue management system.