Dynamic Pricing for Urban Parking Lots

Capstone Report - Summer Analytics 2025

Background & Motivation

Urban parking is a limited, high-demand resource. Static pricing causes underutilization or overcrowding. This project creates a dynamic pricing system that adjusts in real-time based on live occupancy, queue length, traffic, events, and competitor prices.

Data Description

Dataset: 14 parking spaces, 73 days, 18 time points per day. Features: Latitude, Longitude, Capacity, Occupancy, Queue Length, Traffic Condition, Special Day, Vehicle Type.

Tech Stack

- Python
- Pandas, NumPy
- Pathway for real-time streaming
- Bokeh for visualizations
- Google Colab
- GitHub

Pricing Models

Baseline Linear Model:

$P_{t+1} = P_t + \alpha * (Occupancy / Capacity)$

Demand-Based Model:

$Demand = \alpha*(Occupancy/Capacity) + \beta*QueueLength + \gamma*Traffic + \delta*SpecialDay + \epsilon*VehicleTypeWeight$

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

Competitor Pricing

Competitor prices are factored by computing distance to nearby lots using the Haversine formula and comparing to average prices within 500 meters.

Smoothing & Constraints

Prices are constrained between 0.5x and 2x base price ($10) and limited to max $1 change per time step.

Rerouting Logic

If occupancy exceeds 90%, vehicles are rerouted to nearby lots within a defined distance.

Visualization

Bokeh plots show real-time pricing vs competitor prices for transparency.

Key Assumptions

Traffic mapping: low=0.3, medium=0.6, high=1.0.

Vehicle weights: car=1.0, bike=0.5, truck=1.5.

Competitor range: within 500 meters.

Price change: max $1 per step.

Workflow

1. Real-Time Data Stream (Pathway)
2. Preprocessing (Pandas, NumPy)
3. Pricing Models
4. Competitor Logic (Haversine)
5. Smoothing & Constraints
6. Bokeh Visual Output
7. Rerouting if overburdened.

Conclusion

This system demonstrates how dynamic pricing balances demand, maximizes utilization, and optimizes revenue for urban parking lots.