**Dynamic Pricing for Urban Parking Lots**

**Capstone Project - Summer Analytics 2025**  
**Hosted by: Consulting & Analytics Club × Pathway**

**1. Project Overview**

Urban parking is a resource-constrained environment, often plagued by congestion, underutilization, or inefficient pricing. The goal of this project is to develop a dynamic, real-time pricing system for 14 urban parking lots using real-world data and economic logic.

The pricing models are built using only Python libraries like pandas, numpy, and pathway, and include:

* A baseline linear model
* A demand-based pricing function
* A competitive pricing model

These models react to real-time features such as:

* Parking lot occupancy
* Queue length
* Traffic conditions
* Special event indicators
* Vehicle type
* Proximity to competitor lots

**2. Dataset Description**

The dataset comprises records collected over 73 days, each day segmented into 18 time points (8:00 AM to 4:30 PM). Each record includes:

**a. Location Features**

* Latitude and Longitude

**b. Parking Lot Features**

* Capacity (max vehicles)
* Occupancy (current vehicles)
* Queue length

**c. Vehicle Information**

* Vehicle type (car, bike, truck)

**d. Environmental Conditions**

* Traffic congestion (low, average, high)
* Special day indicator

**3. Pricing Models**

**Model 1: Baseline Linear Model**

The price increases linearly with occupancy rate:

Price\_t = BasePrice + α \* (Occupancy / Capacity)

* BasePrice = $10
* α = 5
* Bounded between $5 and $20

**Model 2: Demand-Based Model**

A more dynamic model incorporating multiple features:

Demand = 1.0 \* (Occupancy / Capacity) + 0.5 \* QueueLength - 0.7 \* Traffic + 1.0 \* SpecialDay + 1.2 \* VehicleWeight

Price\_t = BasePrice \* (1 + λ \* NormalizedDemand)

* Normalized demand ensures smooth price transitions
* λ = 0.2
* Output clipped to [$5, $20]

**Model 3: Competition-Aware Model**

This model uses geolocation data to factor in prices of nearby lots:

* If nearby lots (<1km) are cheaper and current lot is over 95% full: decrease price or suggest rerouting
* If nearby lots are more expensive: increase price marginally

Haversine formula is used to compute distances between lots.

**4. Real-Time Simulation**

Pathway is used to simulate streaming data in real-time:

* A custom CSV stream is written line-by-line with delays
* Pathway ingests this stream using schemas
* Average prices are continuously emitted and written to an output CSV

Real-time monitoring is enabled with pw.run(monitoring\_level=ALL).

**5. Visualization**

Bokeh is used for live, interactive plots:

**a. Price Model Comparison**

* A line chart showing all 3 models over time for a given lot

**b. Competitor Pricing**

* A bar chart showing price differences among nearby lots at a timestamp

**6. Rerouting Feature**

If a parking lot exceeds 95% occupancy and nearby lots have space (<80%) within 1km, the system:

* Suggests the best lot based on proximity and pricing

**7. Assumptions**

* Vehicle type weights: Car = 1.0, Bike = 0.6, Truck = 1.5
* Prices are capped between $5 and $20 for realism
* Demand normalization ensures stable pricing
* Traffic levels mapped to [0.2, 0.5, 0.8]

**8. Tools & Stack**

* Python
* Pandas, NumPy
* Pathway (for real-time streaming)
* Bokeh (for plotting)
* Google Colab

**9. Conclusion**

This project demonstrates a full-stack real-time intelligent pricing engine that reacts to various dynamic signals in an urban parking environment. The tiered models reflect increasing sophistication, and the use of Pathway brings real-time simulation into practice.

Future extensions may include:

* Deep learning-based pricing models
* Traffic API integration
* Dynamic rerouting interfaces with maps

**10. Repository**

All working code, visualizations, and this report are present in the linked GitHub repository.

**GitHub Link:** https://github.com/Prashun-Mishra/dynamic-parking-pricing-pathway

**Report by:** Prashun Mishra

**Email:** [mishraprashun47@gmail.com](mailto:mishraprashun47@gmail.com)