Capstone Project Report

Real-Time Dynamic Parking Price Optimization

Introduction

This project builds a real-time streaming pipeline to dynamically compute parking prices based on:

- Real demand factors (occupancy, traffic, queue, etc.)
- Special days and vehicle types
- Competitor prices based on location (using latitude & longitude)

Built using Python, Pathway, Pandas, and visualized live with Bokeh + Panel.

❖ **Model 1: Baseline Daily Pricing** (Link to Colab)

Goal: Calculate a simple daily parking price based only on how much the occupancy fluctuates during the day.

Method:

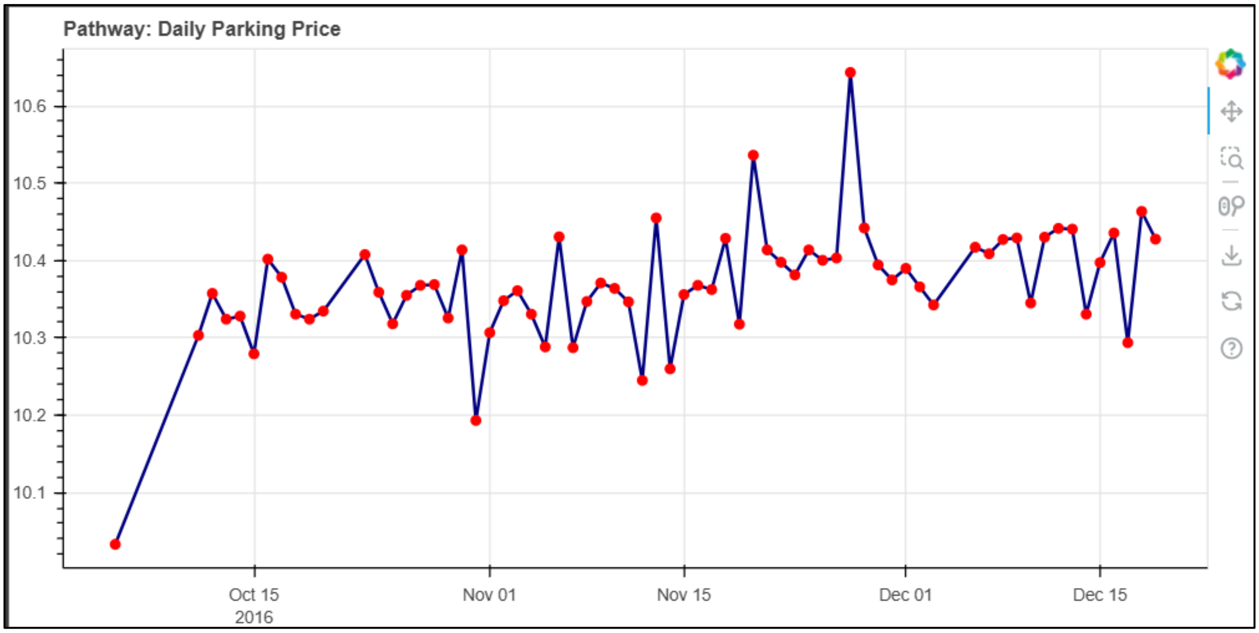
- Use a daily tumbling window.
- For each day, compute:
  - occ_max = highest occupancy
  - occ_min = lowest occupancy
  - cap = capacity
- Calculate price:

price = 10 + (occ_max - occ_min) / cap

Explanation:

- If occupancy fluctuates a lot → price slightly increases.
- Keeps price near base price (10).

➢ **Bokeh Plot for Model 1:**



❖ **Model 2: Demand-Based Dynamic Pricing** (Link to Colab)

Goal: Use real demand features to make the price responsive to actual daily demand patterns.

Method:

- Features:
    - occupancy ratio
    - queue length
    - traffic condition nearby
    - vehicle type
    - special day flag
- Calculate demand with weighted formula:

demand = α·avg occ ratio + β·avg queue - γ·avg traffic + δ·special day + ε·vehicle weight

- Normalize demand to [0, 1].
- Compute price:

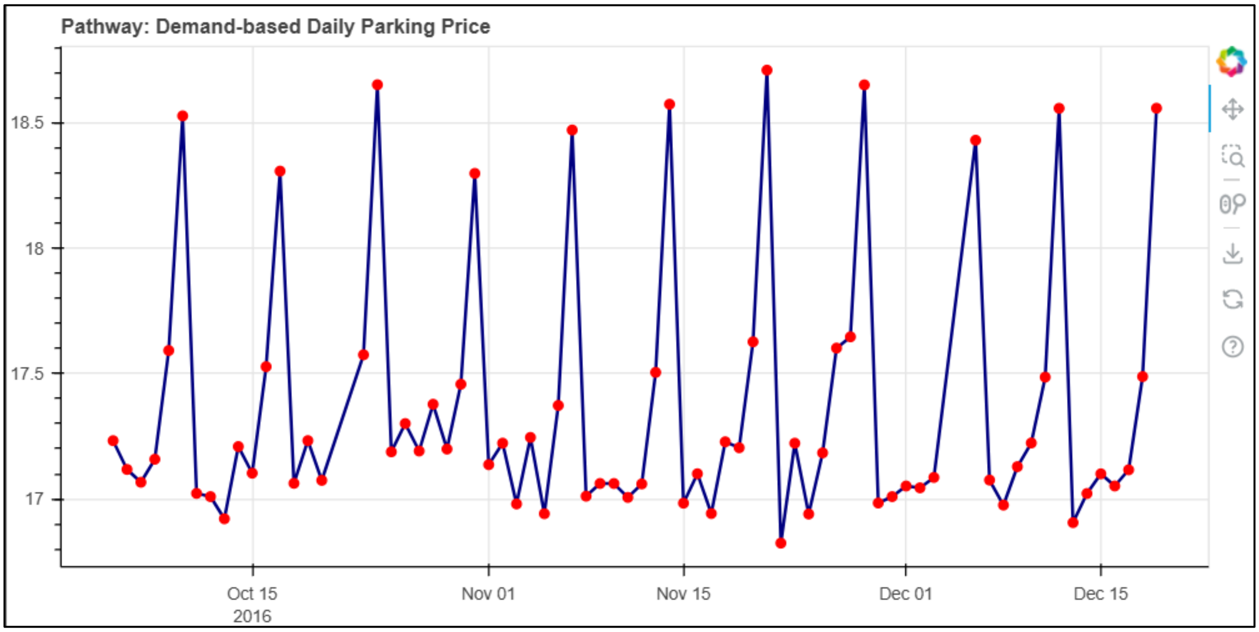
price = 10 × (1 + λ × norm demand)

- Bound price to stay fair.

Explanation:

- Days with high queue, occupancy, and special days → price increases.
- Higher traffic or low occupancy → price stays low.

➢ **Bokeh Plot for Model 2:**



❖ **Model 3: Competitive Dynamic Pricing** (Link to Colab)

Goal: Make the price dynamic not only based on demand, but also by comparing with nearby competitor parking lots.

Method:

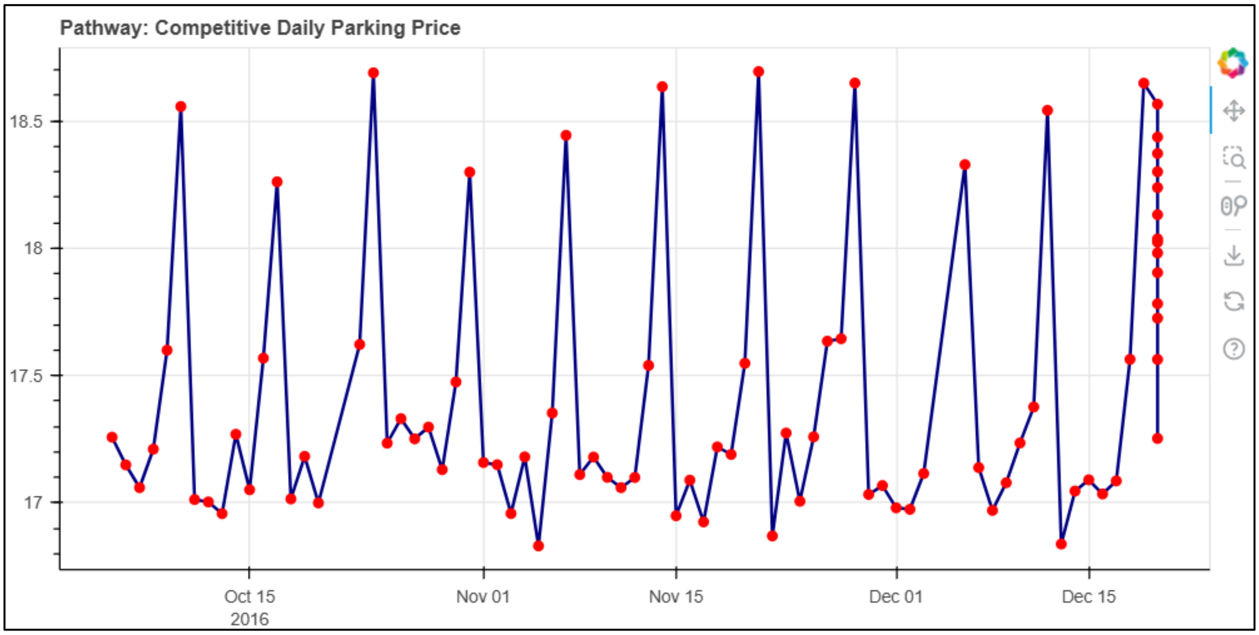
- Use latitude & longitude to precompute nearby lots (within 500 m).
- For each lot and day:
  - compute own price as in Model 2
  - compute average competitor price of nearby lots
- Adjust:

final price = own price - (avg competitor price - base price) × 0.2

Explanation:

- If competitors are cheaper → lower price a bit.
- If competitors are more expensive → keep or increase price.
- Makes pricing smarter & more competitive.

➢ **Bokeh Plot for Model 3:**



✅ Conclusion

- Model 1: very simple, small changes
- Model 2: dynamic demand-based pricing
- Model 3: realistic, competitive pricing using location

All models run live in Pathway, visualized in Bokeh.

Prepared by: Bedabrat Barma, IIT Guwahati, Roll No.: 240103034