**Researching for Optimizing Delhi Metro**

The **Delhi Metro** is the capital of India, carrying over **5** million people every day across the city. It's one of the country’s greatest public transport achievements. But like any large system, it faces many challenges especially during peak hours and during off-peak times, when many compartments run nearly empty. The schedules often remain fixed, even when demand changes throughout the day.

# Techniques used by International Metro Systems

**London Underground**

* It brings together data from **weather forecasts**, **city events**, and even **social media trends** to predict when and where crowds are likely to build up — adjusting train schedules in real time to handle surges during major events or unexpected situations.

**Tokyo Metro**

* Employs AI-powered passenger flow balancing by suggesting alternate routes via mobile apps.

**Hong Kong MTR**

* Implements dynamic pricing to spread passenger loads, offering off-peak discounts to reduce rush hour congestion.

# Challenges in Delhi Metro

1) Overcrowding during peak hours.

2) Specific stops have significantly higher traffic.  
3) Static train schedules that don’t adapt to real-time ridership patterns.  
4) Underutilization of machine learning and real-time optimization.

# Optimization Ideas for Delhi Metro

1. **Dynamic Train Allocation**: Deploying trains where real-time demand is highest.
2. Introduce mobile app features that alert passengers about congestion levels on platforms or in trains, helping distribute passengers more evenly.
3. **Dynamic Pricing:** It offers fare discounts during off-peak hours, encouraging people to travel at less crowded times helping ease the rush during peak hours and making commutes more comfortable for everyone.
4. Identifying underperforming routes and optimizing them.
5. Using trip and timing data to predict and prevent failures.

# Machine Learning Models

**By searching on Internet I got some models which can help in optimizing Delhi Metro:**

* Time-Series Forecasting: Algorithms like Random Forest, XGBoost, or LSTM can predict station-level demand with higher accuracy.
* Use clustering to identify route families with similar demand.
* Apply regression models to estimate travel times and crowd flow.
* Detect anomalies and forecast failure points using classification algorithms.

# How to do

1. Data exploration and visualization
2. Model training and testing
3. Model integration with visualization tools