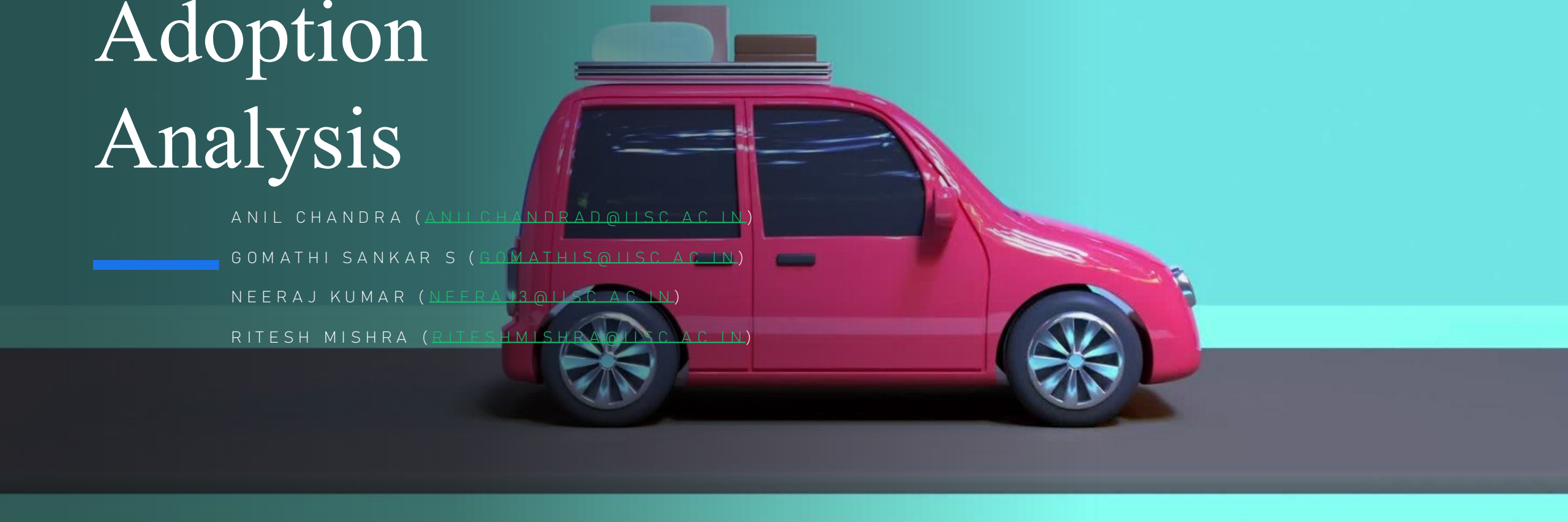
EV vs ICE Vehicle Adoption Analysis

ANIL CHANDRA (ANILCHANDRAD@IISC.AC.IN)

GOMATHI SANKAR S (GOMATHIS@IISC.AC.IN)

NEERAJ KUMAR (NEERAJ3@IISC.AC.IN)

RITESH MISHRA (RITESHMISHRA@IISC.AC.IN)



Agenda

- Problem Statement & Objectives
- Data Collection & Preprocessing
- Infra (HDFS, PySpark)
- Exploratory Data Analysis (EDA)
- Machine Learning Models
  - Linear Regression
  - GBT
  - Random Forest
- Insights & Recommendations
- Conclusion

Problem Statement

- Objective: Compare Electric Vehicles (EV) vs Internal Combustion Engine (ICE) vehicles
- Analyze adoption trends across 1600+ Indian cities over 120 months (2015-2024)
- Key Questions:
  - What factors drive EV adoption?
  - Can we predict EV adoption probability?
  - Which cities show highest EV adoption?
  - What are the temporal trends?
- Dataset: 200000 records with temporal and geographic features

Motivation

Environmental Concerns: Rising air quality index (AQI) in urban regions and increasing CO_2 emissions

Economic Factors: Escalating fuel prices and dependence on fossil fuels

Policy Initiatives: Government incentives, subsidies, and EV charging infrastructure development

Consumer Hesitation: Range anxiety, charging delays, and higher upfront costs creating adoption barriers

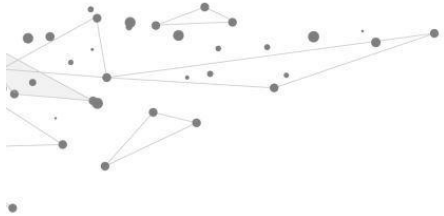
Data-Driven Insights: Need for predictive models to inform policy decisions and infrastructure planning

Design Goals

- **Predictive Accuracy:** Build models with ~80% R^2 score
- **Scalability:** Handle large-scale datasets (200K+ records) efficiently using distributed computing
- **Comprehensive Analysis:** Implement multiple ML paradigms (random forest, gradient boost, linear regression)
- **Feature Understanding:** Identify and quantify key drivers of EV adoption
- **Actionable Insights:** Generate city segmentation and trend forecasts for policy planning

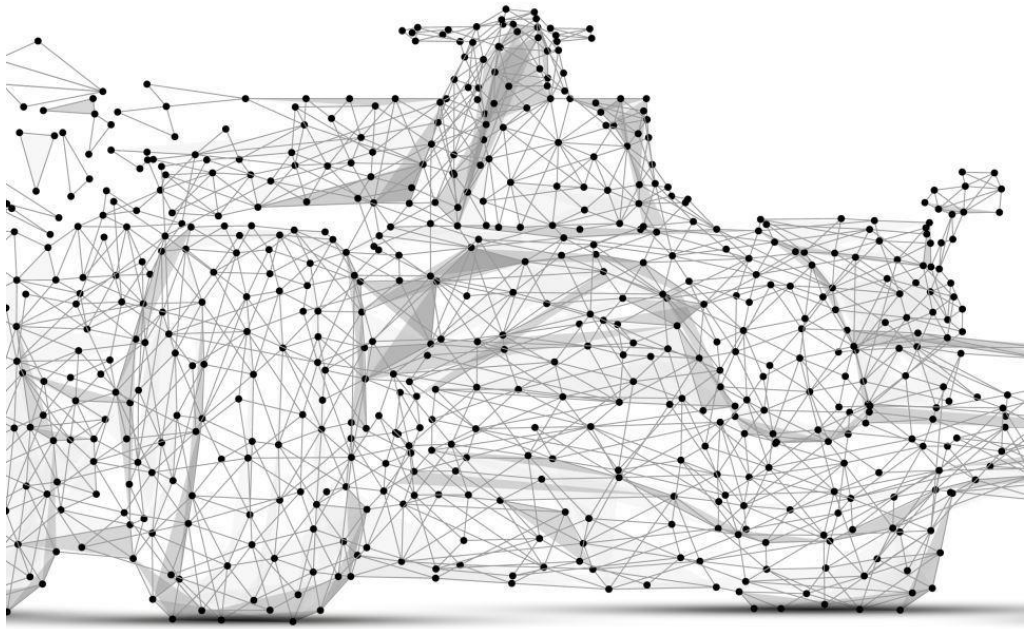
How we created dataset?

- The EV–ICE dataset was created synthetically but designed to behave like **real Indian automotive market data**.
- **10 years of monthly data** (2015–2024)
- **1,667+ cities & taluks** across India
- **200,000+ rows**
- **30+ features + engineered attributes**
- Real-world dynamics were embedded, including:
  - COVID waves (2020, 2021, 2022)
  - Festival surges (Diwali, Dasara, Pongal, Holi)
  - Natural calamity dips
  - Fiscal year cycles
  - Policy-driven subsidy boosts
  - Noise, missing values, and outliers



About Dataset

- The dataset captures the **economic, environmental, infrastructural, and behavioural indicators** that influence EV adoption in India. It includes:
- Fuel price fluctuations
- Charging infrastructure status
- City-level AQI & pollution
- Awareness & behavioural scores
- EV & ICE sales patterns
- Subsidies, government investments
- Vehicle pricing & maintenance patterns



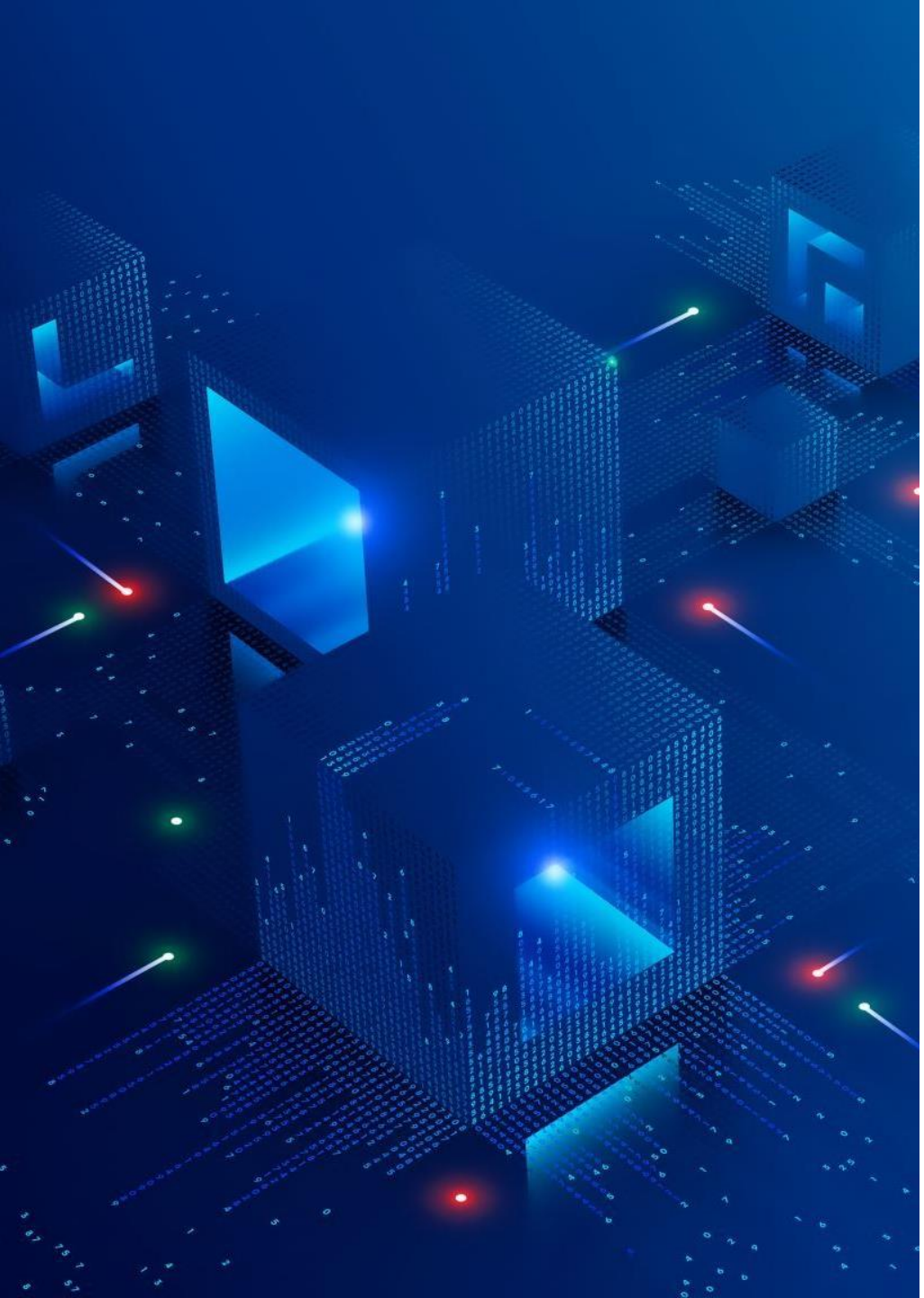
Dataset Overview

Column	*Description*
fuel_price_per_litre	Monthly fuel cost
avg_city_aqi	Air quality indicator
charging_stations_per_10km	Infra density
ev_subsidy_amount	Govt subsidy on EV
income_level	Avg income per region
vehicle_price_ev / ice	EV & ICE pricing
battery_range_km	EV range
charging_time_minutes	Time to charge fully
maintenance_cost_ev / ice	Maintenance cost
ev_awareness_score	Awareness level
city_ev_readiness_index	Composite readiness score
ev_sales_last_year	Last year EV sales
ice_sales_last_year	ICE sales
co2_emission_city	Pollution level
population_density	People per sq.km
gov_infra_investment_crores	Infra investment
charging_infra_growth_rate	Infra growth speed
consumer_range_anxiety_score	Fear of low range
vehicle_resale_value_ev / ice	Resale values
charging_cost_per_full_charge	Charging cost

About the Target Column

Target: `ev_adoption_probability`

- A continuous value (0–1) representing the **likelihood that a city/taluk in a given month prefers EV over ICE**.
- It was computed using:
  - Charging infra
  - Subsidy amount
  - Awareness score
  - Battery range
  - Charging time
  - Other normalized growth indicators



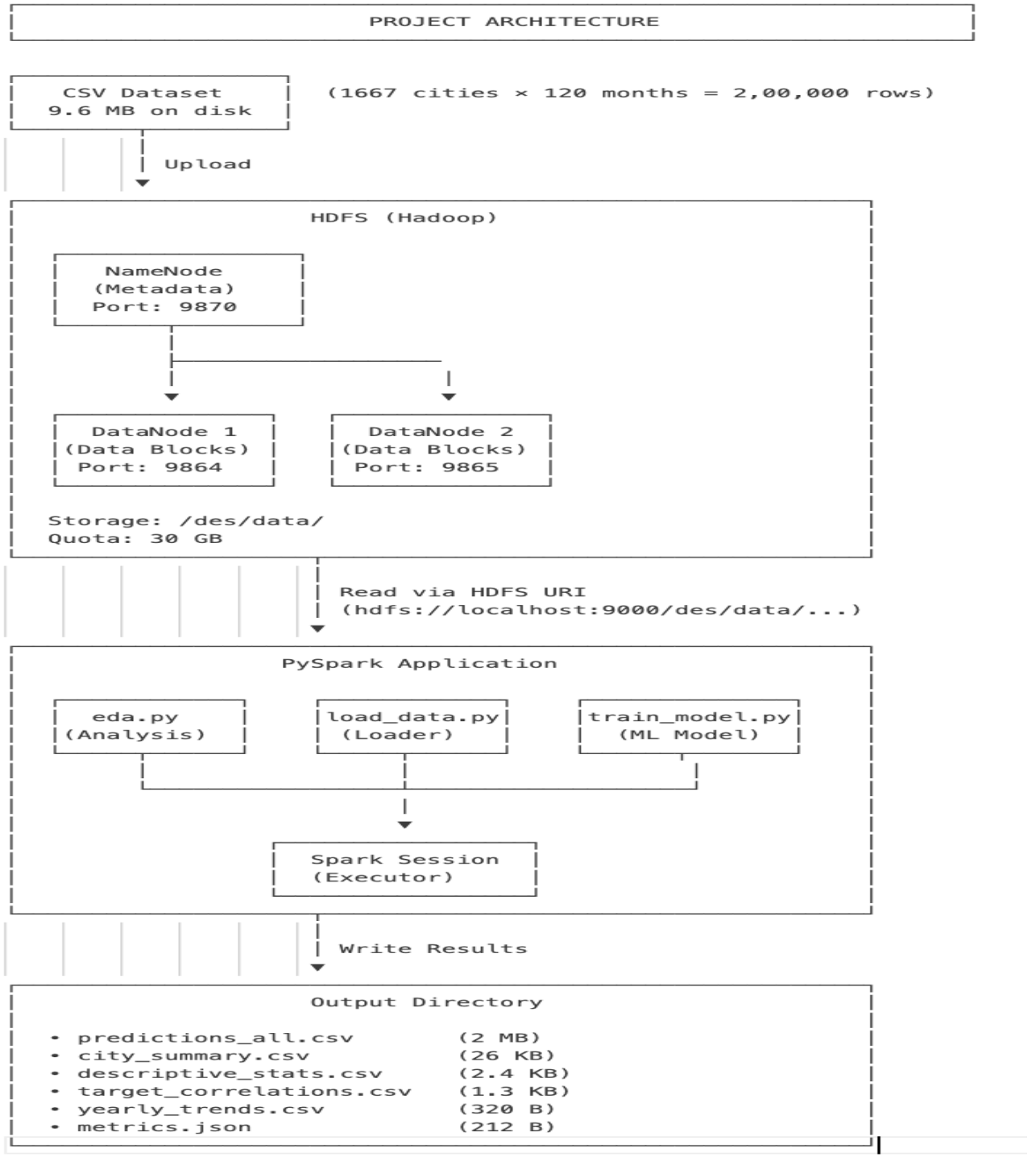
High Level Design

Data Ingestion (HDFS) → Preprocessing & EDA → Feature Engineering →Model Training → Evaluation → Deployment → Visualization

Key Components:

- 1. **Data Layer:** CSV file storage with Spark DataFrame abstraction
- 2. **Processing Layer:** PySpark for distributed data processing
- 3. **Exploratory Data Analysis (EDA)**
- 4. **ML Layer:** Spark ML for scalable machine learning
- 5. **Visualization Layer:** Matplotlib for exploratory data analysis
- 6. **Evaluation Layer:** Multiple metrics (RMSE, R^2, MAE)

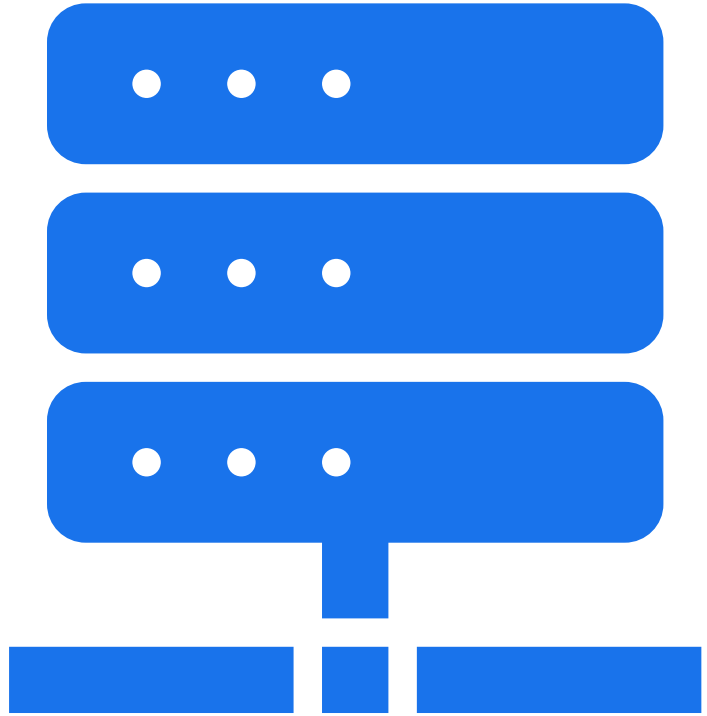
Architecture Diagram



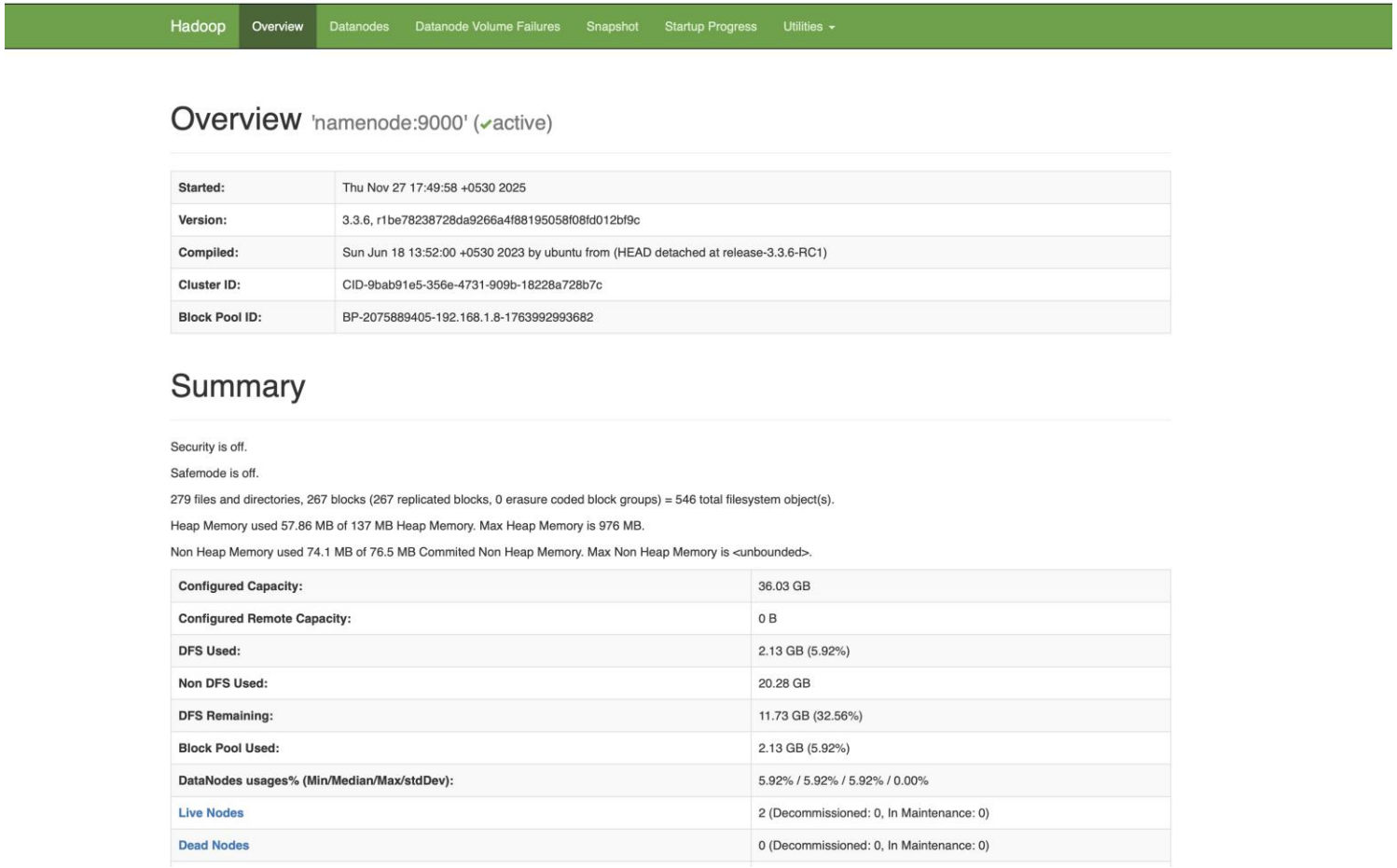
HDFS

- 3-Node Cluster
- NameNode (namenode): 4GB / 2 cores / 40 GB
- DataNode 1 (datanode1): 4GB / 2 cores / 40 GB
- DataNode 2 (datanode2): 4GB / 2 cores / 40 GB

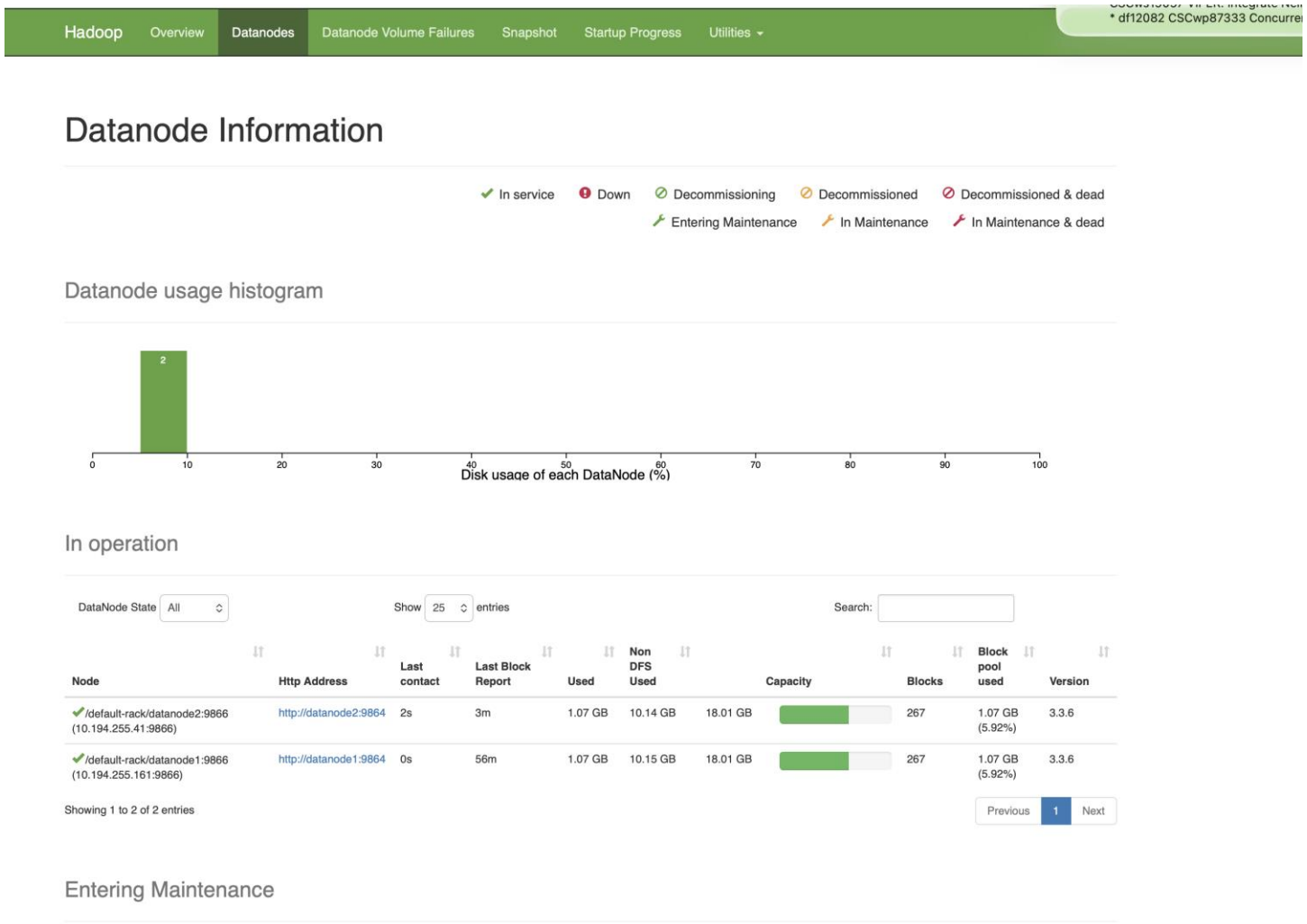
- Monitoring
- NameNode UI → http://namenode:9870



Namenode


Hadoop
Overview
Datanodes
Datanode Volume Failures
Snapshot
Startup Progress
Utilities
Overview 'namenode:9000' (✔active)
Started: Thu Nov 27 17:49:58 +0530 2025
Version: 3.3.6, r1be78238728da9266a4f88195058f08fd012bf9c
Compiled: Sun Jun 18 13:52:00 +0530 2023 by ubuntu from (HEAD detached at release-3.3.6-RC1)
Cluster ID: CID-9bab91e5-356e-4731-909b-18228a728b7c
Block Pool ID: BP-2075889405-192.168.1.8-1763992993682
Summary
Security is off.
Safemode is off.
279 files and directories, 267 blocks (267 replicated blocks, 0 erasure coded block groups) = 546 total filesystem object(s).
Heap Memory used 57.86 MB of 137 MB Heap Memory. Max Heap Memory is 976 MB.
Non Heap Memory used 74.1 MB of 76.5 MB Commited Non Heap Memory. Max Non Heap Memory is <unbounded>.
Configured Capacity: 36.03 GB
Configured Remote Capacity: 0 B
DFS Used: 2.13 GB (5.92%)
Non DFS Used: 20.28 GB
DFS Remaining: 11.73 GB (32.56%)
Block Pool Used: 2.13 GB (5.92%)
DataNodes usages% (Min/Median/Max/stdDev): 5.92% / 5.92% / 5.92% / 0.00%
Live Nodes 2 (Decommissioned: 0, In Maintenance: 0)
Dead Nodes 0 (Decommissioned: 0, In Maintenance: 0)

Datanode


Hadoop
Overview
Datanodes
Datanode Volume Failures
Snapshot
Startup Progress
Utilities
Datanode Information
In service
Down
Decommissioning
Decommissioned
Decommissioned & dead
Entering Maintenance
In Maintenance
In Maintenance & dead
Datanode usage histogram
2
0
10
20
30
40
50
60
70
80
90
100
Disk usage of each DataNode (%)
In operation
DataNode State All
Show 25 entries
Search:
Node
Http Address
Last contact
Last Block Report
Used
Non DFS Used
Capacity
Blocks
Block pool used
Version
/default-rack/datanode2:9866 (10.194.255.41:9866)
http://datanode2:9864
2s
3m
1.07 GB
10.14 GB
18.01 GB
267
1.07 GB (5.92%)
3.3.6
/default-rack/datanode1:9866 (10.194.255.161:9866)
http://datanode1:9864
0s
56m
1.07 GB
10.15 GB
18.01 GB
267
1.07 GB (5.92%)
3.3.6
Showing 1 to 2 of 2 entries
Previous
1
Next
Entering Maintenance

EDA

- Null value handling
- Outlier removal
- Skewness correction
- Data visualization techniques
- Feature extraction
- Noise removal

Feature Comparisons

More chargers → higher EV adoption

Higher subsidies → higher adoption

Higher fuel prices → more EV preference

Higher awareness → strong EV push

Longer battery range → reduces range anxiety

Lower charging time → increases adoption

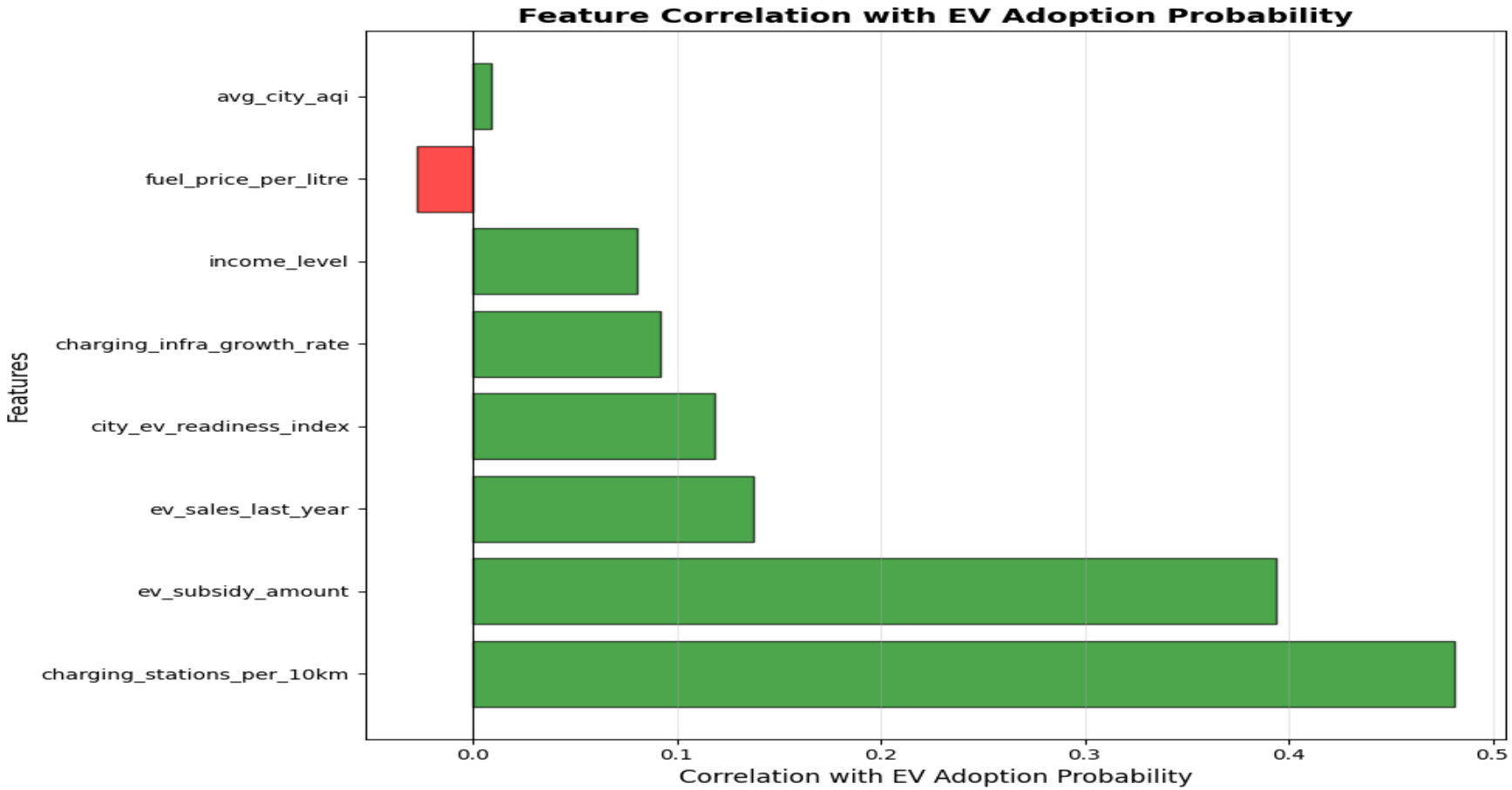
Lower electricity cost → lowers operational cost → boosts adoption

Higher congestion/AQI → pushes people toward EVs

Feature Extractions

- Extracted year & month from Date (DD-MM-YYYY)
- City_Index and State_Index using StringIndexer
- Affordability Index: (Price / Income)
- Infrastructure Confidence: (Range * Station Density)
- Savings Ratio (Fuel Price / Elec Price)
- Resale Confidence (EV Resale Value / ICE Resale Value)
- Market Momentum (EV Sales / ICE Sales)

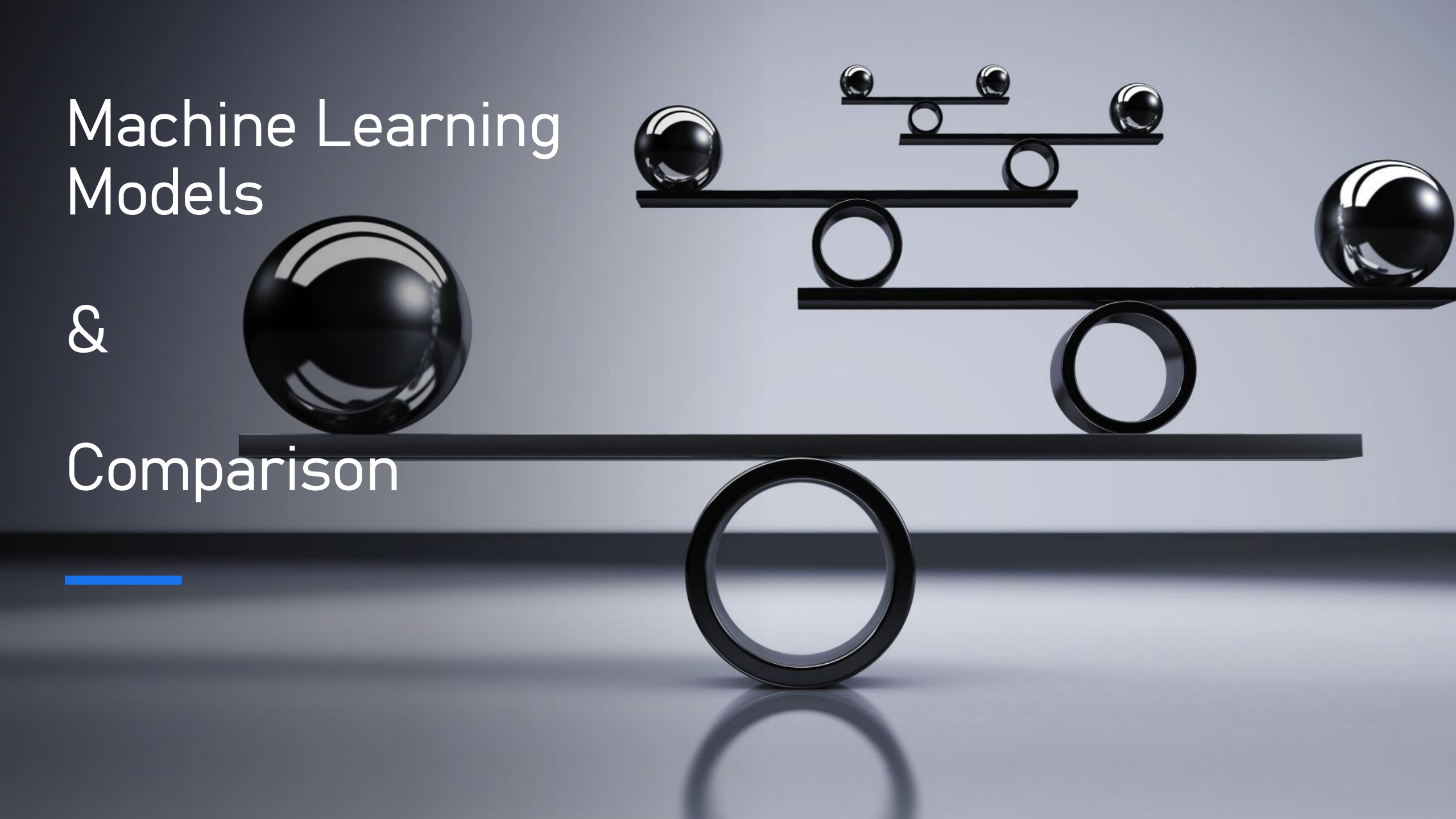
Visualization



Machine Learning Models

&

Comparison



Why Regression?

(Why not Classification/Clustering)

Our output variable (EV adoption probability) is continuous (0–1 range), not categorical, so regression fits naturally

Classification requires discrete labels (High/Medium/Low adoption) — but we wanted exact prediction, not grouping

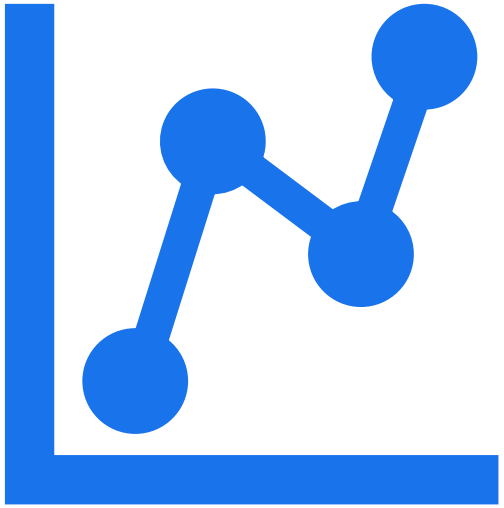
Clustering only groups similar cities — it cannot predict future adoption values

Regression allows us to quantify numerical influence of each factor (price, AQI, infrastructure etc.)

Helps in policy decision-making with measurable outputs instead of broad classes

Regression

- **Linear Regression**
- **Random Forest Regressor**
- **Gradient Boosted Trees (GBT)**



Linear Regression

Why?

- Baseline model to understand linear relationships
- Faster to train & tune

Hyperparams	Value	Usage
maxIter	100	More Optimization Cycles
regParam	0.01	Control overfitting
elasticNetParam	0.0	Pure Ridge Regression Mode

Random Forest

Why?

- Handle non-linear & complex patterns
- Work with large multi-feature data
- Reduces overfitting by using Bagging

Hyperparams	Value	Usage
numTrees	100	More stability & accuracy
maxDepth	10	Control overfitting & tree growth
maxBins	2000	Finer feature splits for better decisions
seeds	42	Ensures reproducible results

Gradient Boosted Trees

Why?

- Learns sequentially & error correction at each step
- For gradual improvement learning
- Works strongly when features are cleaned

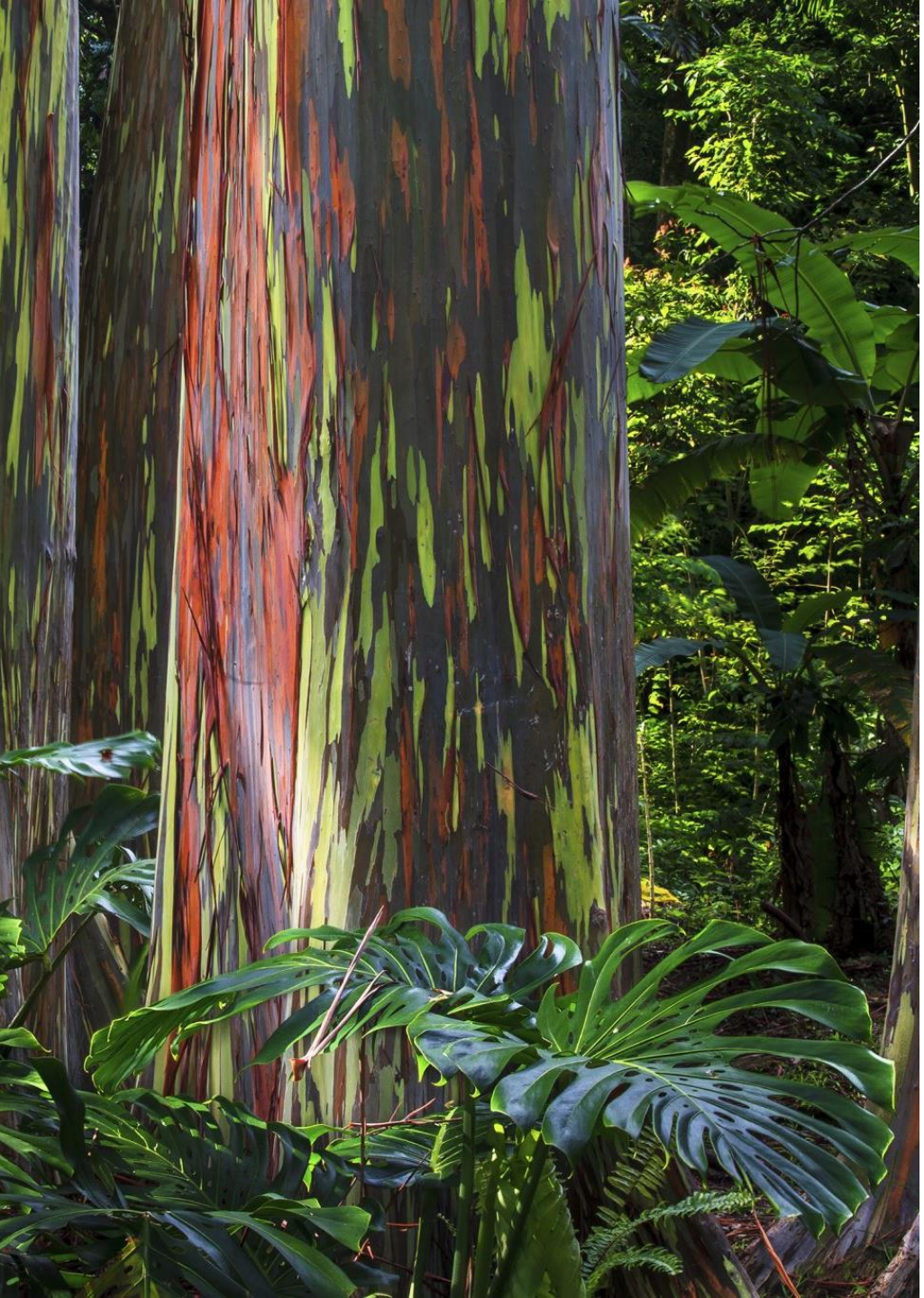
Hyperparams	Value	Usage
maxIter	100	More boosting stages & better refinement
maxDepth	5	Prevents overfitting
maxBins	2000	Better split for high cardinality features
seeds	42	Ensures reproducible results

ML Performance Evaluation Without EDA

Model	RMSE	R^2 Score	MAE
Random Forest	0.0510	0.7414	0.0116
Gradient Boosted Trees	0.0586	0.7049	0.0131
Linear Regression	0.0649	0.6218	0.0265

ML Performance Evaluation With EDA

Model	RMSE	R^2 Score	MAE
Random Forest	0.0510	0.7914	0.0116
Gradient Boosted Trees	0.0586	0.7249	0.0131
Linear Regression	0.0612	0.6702	0.0245

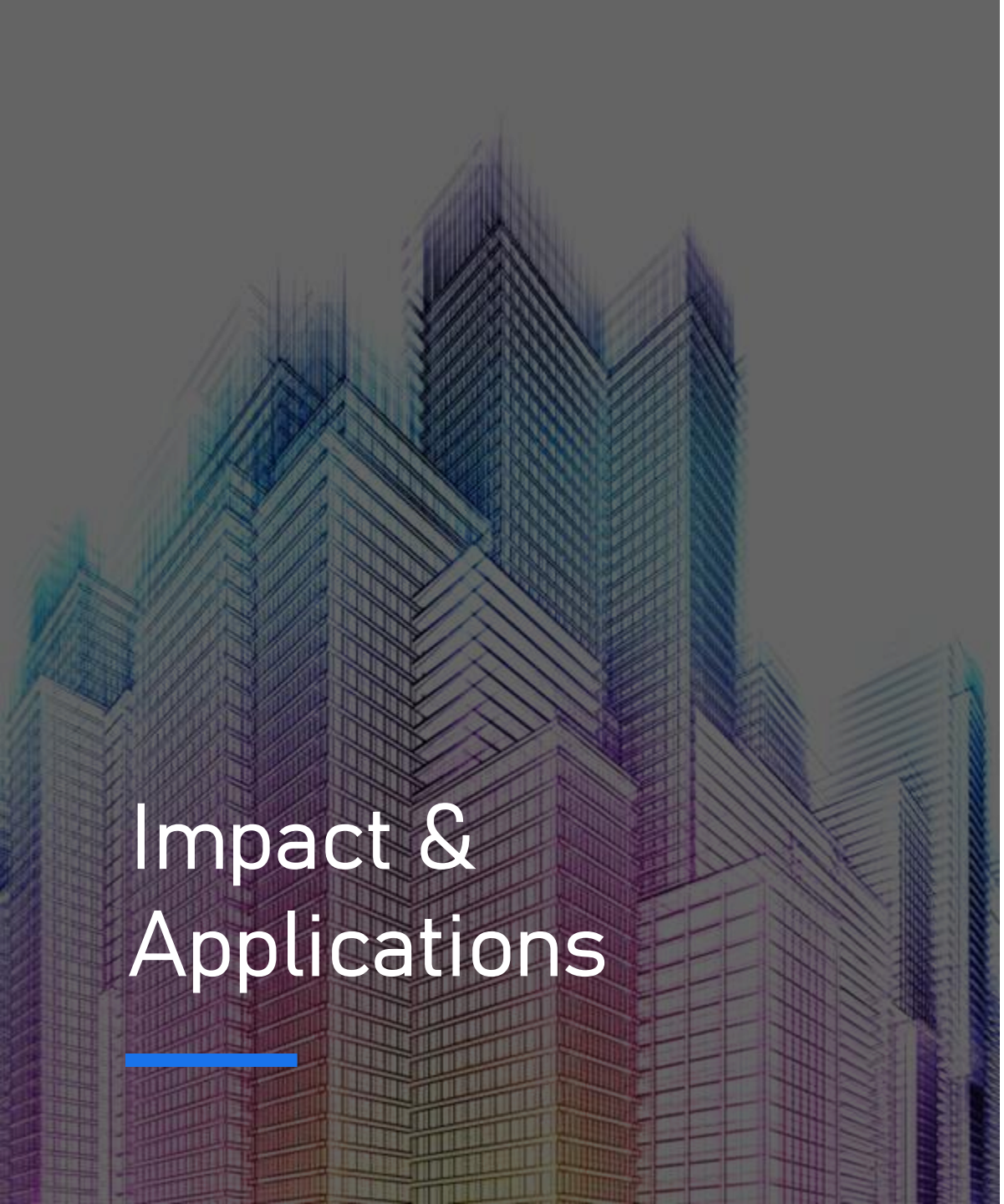


Model Comparison Summary

- Random Forest achieved the best R^2 (~0.79), lowest RMSE & MAE - overall top performer
- Gradient Boosted Trees delivered the second-best results with ~0.72 R^2 after EDA
- Linear Regression trailed behind with ~0.67 R^2 - works as baseline reference only
- Tree-based models clearly outperform linear models due to non-linear EV adoption behavior
- *"Random Forest is most suitable for EV adoption prediction"*

Future Extensions

- Scale to national coverage, apply advanced models (GNNs)
- Expand dataset using real charging station, vehicle sales & policy data for higher prediction reliability
- Build a full production MLOps system.
- Create EV Adoption Heatmap Dashboard for policymakers & automotive industry planning



Impact & Applications

- Policy
  - Identify high-potential cities for EV infrastructure.
  - Design localized incentive programs.
  - Forecast charging-station demand.
  - Optimize subsidy allocation.
- Industry
  - EV market segmentation and demand forecasting.
  - Supply-chain and production planning.
  - Competitive benchmarking.
  - Investment risk assessment.
- Research
  - Baseline for comparative studies.
  - Big-data / ML education case study.
  - Dataset for ML benchmarking.

Conclusion

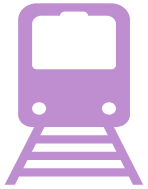
Project Summary

Processed 2,00,000 **records** using **regression**

79.24% R^2 in regression (exceeded 75% target).

51.75% EV growth identified over 10 years.

92.09% city dominance in feature importance.



Insights

Highlights need for **localized policies & infrastructure planning**.

City-specific factors are the strongest drivers of EV adoption.

Project Impact

Enables **data-driven decision-making** for policymakers, industry, and researchers.

Supports **sustainable transportation planning** in India.

Thank you :)