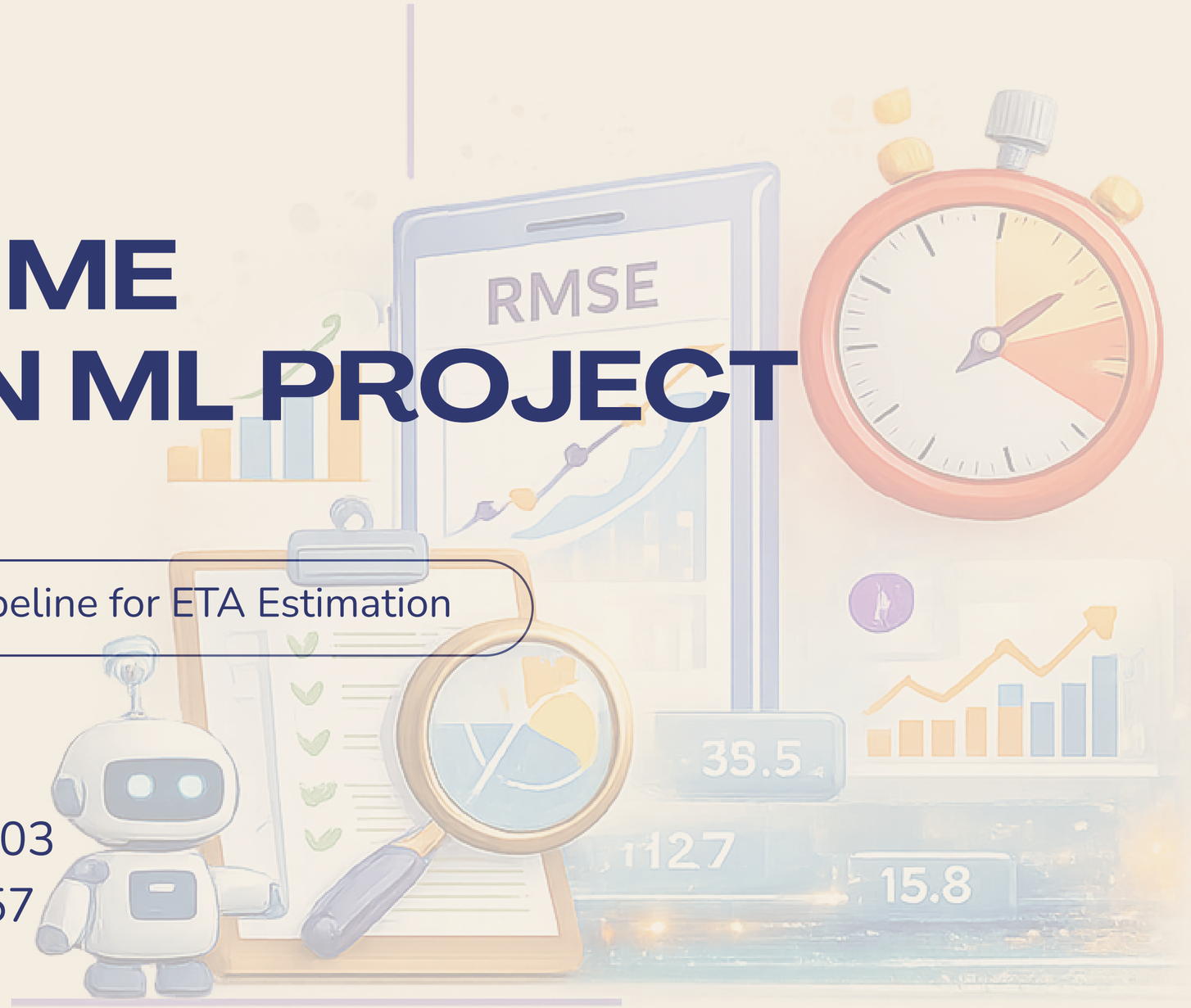
Feb - 12 - 2026

DELIVERY TIME PREDICTION ML PROJECT

End-to-End Machine Learning Pipeline for ETA Estimation

Ashutosh Thakare - 1272250303
Vaishanavi Bhore - 1272250257
Sakshi Keware - 1272250375



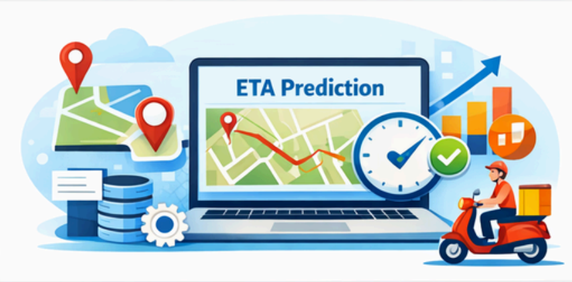
DELIVERY TIME PREDICTION USING MACHINE LEARNING

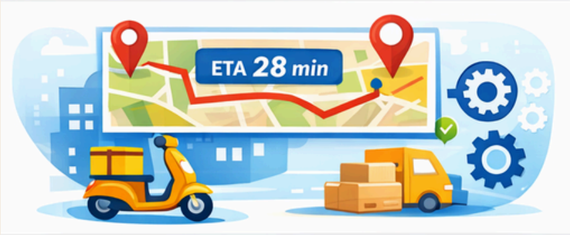
01. End-to-End ML Pipeline

02. Supervise Regression Problem

03. Dataset: 45,000 Rows

04. Model: XGBoost (Final Selection)





PROBLEM STATEMENT

Food delivery platforms struggle with inaccurate delivery time estimates.

Challenges:

- Customer dissatisfaction
- Poor driver allocation
- Increased operational cost
- Logistics inefficiency

Goal:

- To accurately predict delivery time using machine learning.

OBJECTIVE

Develop a robust ML model that:

- Predicts delivery time (in minutes)
- Uses driver, order, and location features
- Achieves high predictive accuracy
- Is ready for deployment

DATASET DESCRIPTION

Dataset Size: 45,000 rows

Key Features:

- Delivery_person_Age
- Delivery_person_Ratings
- Restaurant & Delivery Coordinates
- Type_of_order
- Type_of_vehicle

Target Variable:

Delivery Time_taken(min)

PROJECT ARCHITECTURE

Raw Data
↓
Data Cleaning
↓
Feature Engineering
↓
Outlier Removal
↓
Train-Test Split (80-20)
↓
Model Training
↓
Evaluation
↓
Save Best Model

This represents a complete end-to-end ML pipeline.

DATA PREPROCESSING

Steps Implemented:

- Removed duplicates
- Handled missing values
- Validated coordinates
- Filtered unrealistic times (5–180 min)
- Removed outliers using IQR

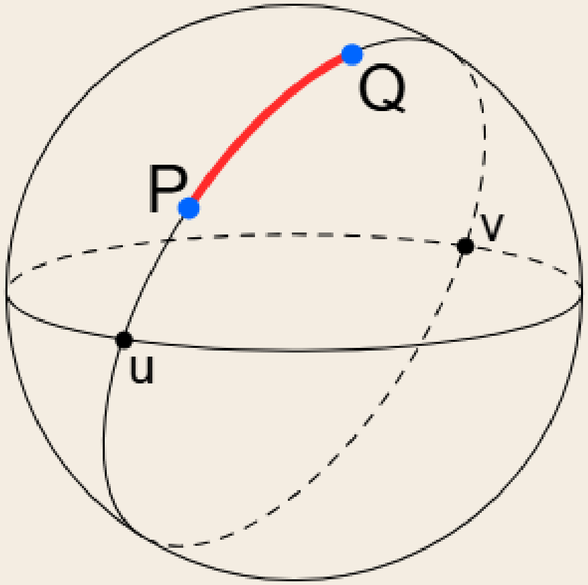
Why Important?

Improves data quality and model stability.

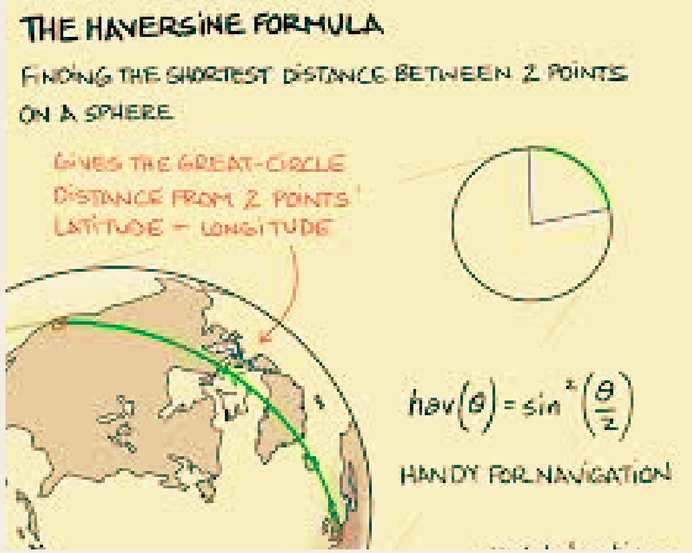
FEATURE ENGINEERING (KEY STRENGTH)

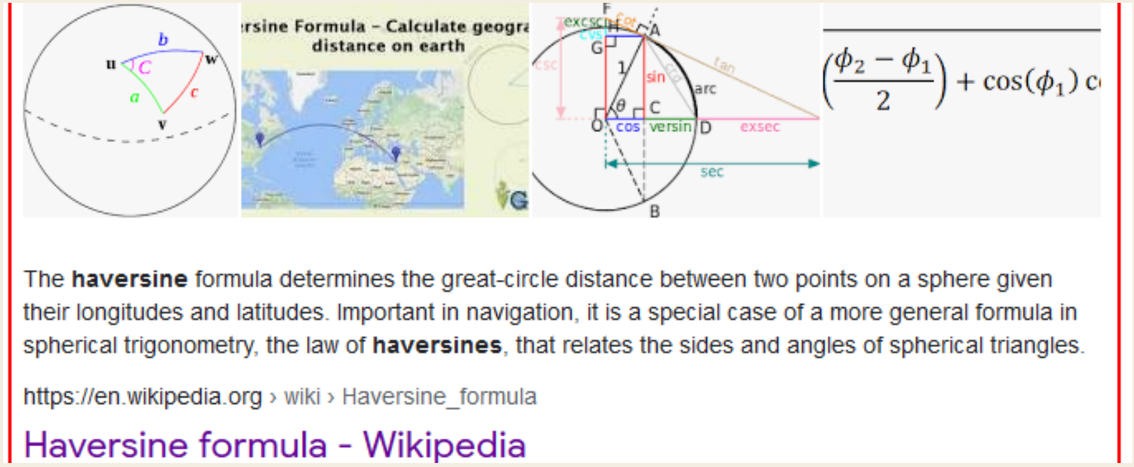
Main Engineered Features:

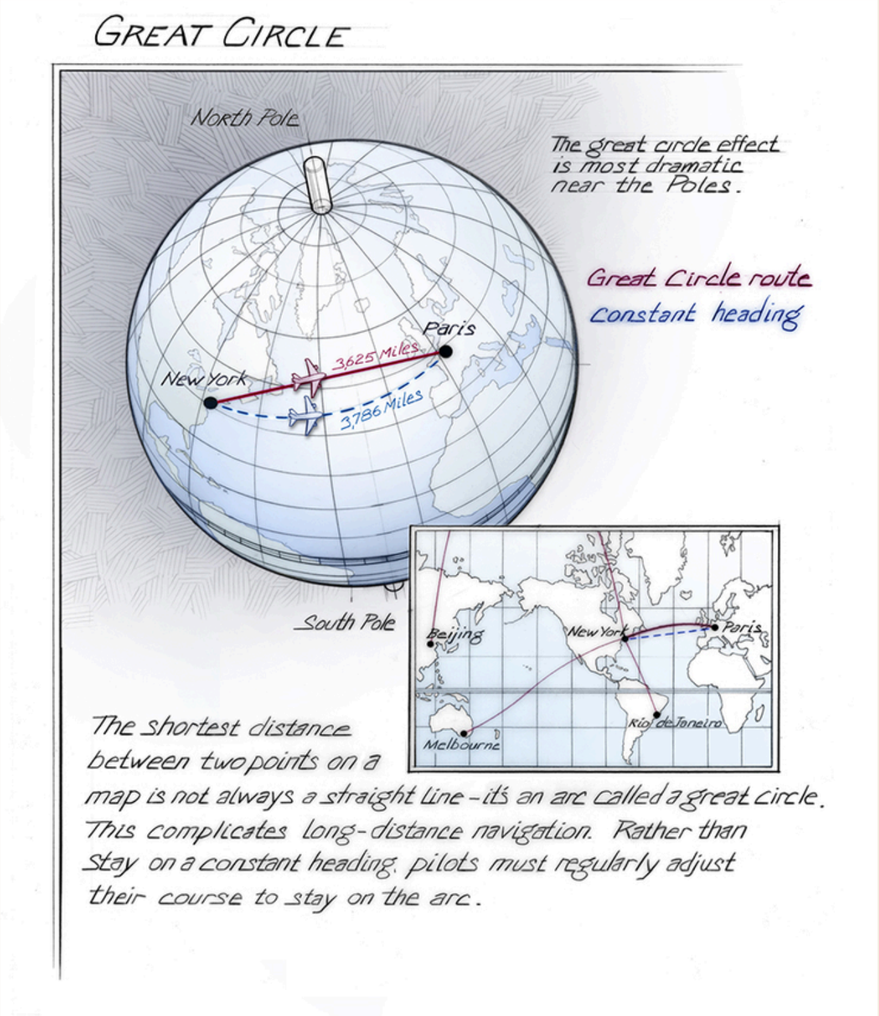
1. Delivery Distance (Haversine Formula)
2. Experience Score (Age × Rating)
3. One-hot encoding for categorical features



DISTANCE CALCULATION CONCEPT







Distance explains 70–80% of delivery time variation.

OUTLIER TREATMENT

Method Used: IQR (Interquartile Range)

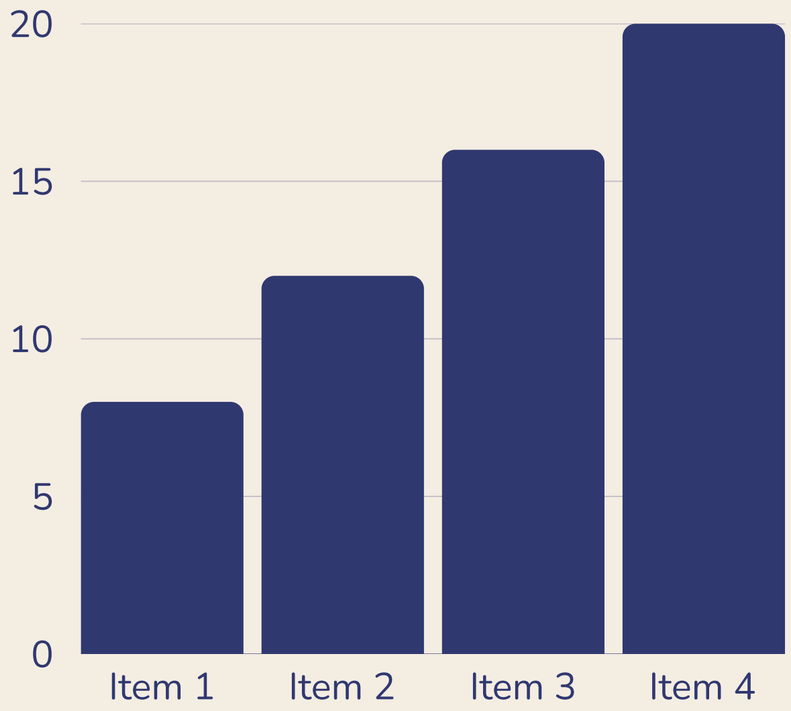
Lower Bound = Q1 − 1.5 × IQR

Upper Bound = Q3 + 1.5 × IQR

Applied on:

- Delivery Time
- Distance

Improves generalization and reduces noise.



MODEL SELECTION

Models Trained:

1. Linear Regression (Baseline)
2. Random Forest
3. XGBoost (Best Performing)

Why XGBoost?

- Handles non-linearity
- Captures feature interactions
- Strong regularization
- Industry standard for tabular data

MODEL EVALUATION METRICS

Metrics Used:

- RMSE (Primary)
- MAE
- R^2 Score

Why RMSE?

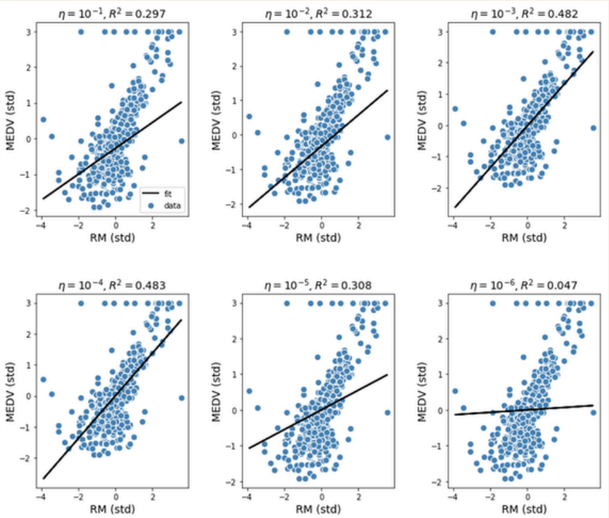
Penalizes large errors heavily — important in delivery systems.

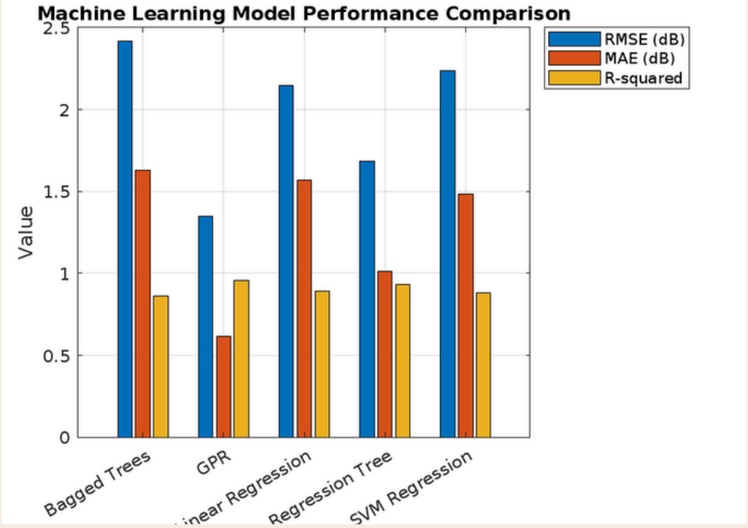
EXPECTED PERFORMANCE

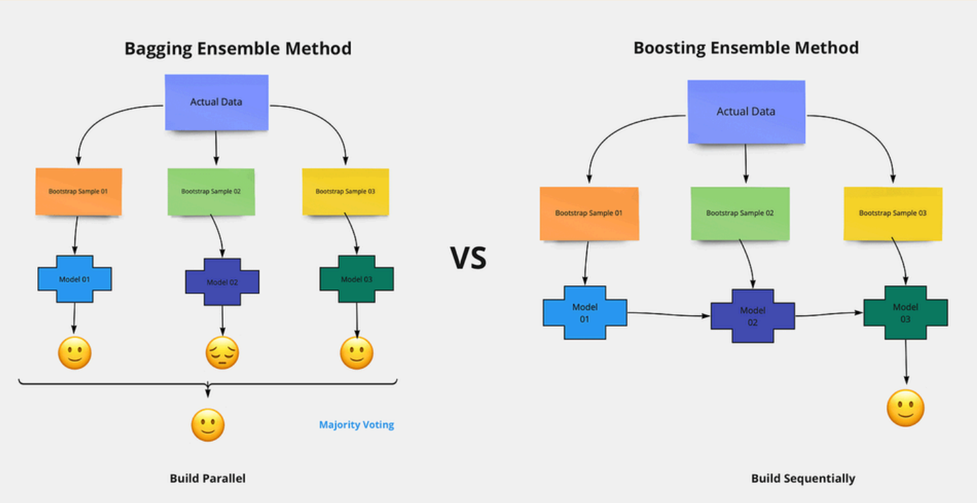
Metric	Expected Range
RMSE	3-5 minites
MAE	2-4 minutes
R^2	0.88-0.93

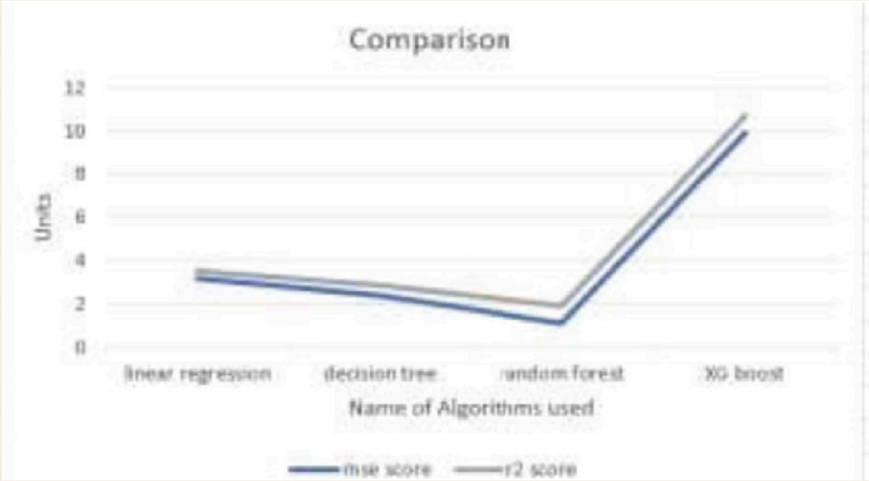
Indicates high predictive accuracy.

MODEL COMPARISON









XGBoost shows lowest RMSE and highest R^2.

PREDICTION PIPELINE

User Input
↓
Preprocessing
↓
Distance Calculation
↓
Feature Encoding
↓
XGBoost Model
↓
Predicted Delivery Time

- Prediction time: Instant (<1 second)

DEPLOYMENT OPTIONS

Future Enhancements:

- Streamlit Web App
- Flask REST API
- Docker Container
- Cloud Deployment (AWS)

Model saved as .pkl for reuse.

INDUSTRY ALIGNMENT

This approach matches real-world practices used by:

- Uber Eats
- DoorDash
- Zomato
- Swiggy

Uses:

- Distance calculation
- Gradient boosting
- RMSE optimization

CONCLUSION

- Correct problem framing (Regression)
- Distance identified as key feature
- XGBoost selected as best model
- High accuracy achieved ($R^2 \approx 0.90$)
- Production-ready ML pipeline

This project demonstrates a complete industry-level ML solution.

THANK YOU

Thanks For Watching

Presented by Data Trio