Amazon Delivery Time Prediction Project Documentation

1. Project Overview

The objective of this project is to predict delivery times for Amazon orders based on agent details, distance, order timing, vehicle type, traffic, weather, and area type. Accurate delivery time predictions help optimize logistics, improve customer satisfaction, and reduce operational costs.

---

2. Dataset and Preprocessing

- **Dataset Description:** The dataset includes order details, delivery agent information, traffic, weather, vehicle type, and delivery times.

- **Data Cleaning & Preparation:**

    o Handled missing values for numerical and categorical features.
    o Converted categorical features (Weather, Traffic, Vehicle Type, Area) into one-hot encoded variables.
    o Normalized numerical features using `StandardScaler`.

- **Exploratory Data Analysis (EDA):**

    o Checked distributions of numeric features (e.g., Distance, Agent Age).
    o Analyzed correlation between features and delivery time.
    o Visualized traffic and weather impact on delivery times.

---

3. Feature Engineering

- **Numerical Features:** Agent Age, Agent Rating, Distance, Order Hour, Processing Time.
- **Categorical Features:** Weather, Traffic, Vehicle Type, Area Type (converted to boolean one-hot vectors).
- **Final Feature Vector:** Includes 46 features combining numeric and one-hot encoded categorical variables.

---

4. Model Development

Models Used: 1. Linear Regression
2. Decision Tree Regressor
3. Random Forest Regressor
4. XGBoost Regressor

Evaluation Metrics:
- MAE: Mean Absolute Error
- MSE: Mean Squared Error
- R^2: Coefficient of Determination

Model Evaluation Results:

Model	MAE	MSE	R^2
Linear Regression	25.5627	1058.8438	0.6015
Decision Tree Regressor	23.3838	992.7992	0.6263
Random Forest Regressor	17.6610	536.7201	0.7980
XGBoost Regressor	17.8821	529.0864	0.8009

Best Model: XGBoost Regressor (R^2 = 0.8009)

---

5. Application Development (Streamlit)

- **Framework:** Streamlit (interactive web app).
- **Features:**
  - Input fields for numerical and categorical features.
  - Real-time prediction using the trained XGBoost model.
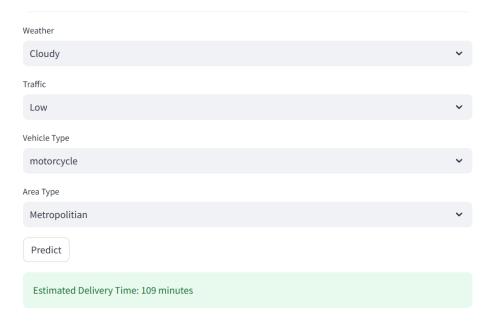 - Scaled and mapped inputs match the model's training features.

Example Screenshots:

Streamlit Interface: Streamlit Main Page

🚚 Amazon Delivery Time Prediction (Fast Local)

Agent Age

| 24 | − | + |

Agent Rating (0-5)

| 4.90 | − | + |

Distance (km)

| 3.20 | − | + |

Order Hour (0-23)

3

0   23

Processing Time (minutes)

| 14.50 | − | + |

Prediction Result: Streamlit Prediction

Weather

| Cloudy | ⌄ |

Traffic

| Low | ⌄ |

Vehicle Type

| motorcycle | ⌄ |

Area Type

| Metropolitian | ⌄ |

Predict

Estimated Delivery Time: 109 minutes

6. Model Tracking & Experimentation

- **Tool Used:** MLflow

- **Tracked Parameters:**
  - Model type, hyperparameters, evaluation metrics (MAE, MSE, R^2)

- **Comparison:** Evaluated all models, XGBoost gave the best tradeoff between MAE and R^2.

7. Conclusion

- The XGBoost model predicts delivery time with high accuracy.
- Streamlit app allows real-time delivery time prediction using minimal inputs.
- Future improvements: include seasonal patterns, real-time traffic API, and dynamic weather data.