Project Report: TrafficTelligence - Traffic Volume Prediction System

---

1. Project Title:
TrafficTelligence - An ML-Powered Web Application for Predicting Traffic Volume

---

2. Objective:
To develop a machine learning-based web application that predicts traffic volume based on weather and time-related features to help urban planners, traffic managers, and commuters make informed decisions.

---

3. Tools & Technologies Used:

- **Programming Languages:** Python, HTML, CSS
- **Libraries:** Scikit-learn, NumPy, Joblib, Flask
- **Frontend:** HTML5, CSS3
- **Backend:** Flask (Python Microframework)
- **Model Serialization:** Joblib

---

4. Dataset Overview:

- Source: Google Drive
- Number of Rows: 48204
- Features Used (Total: 19):
    - Numerical: Temperature, Rain, Snow
    - Time: Hour, Month, Day, Weekday
    - Categorical/Binary: Is_weekend, Is_holiday
    - One-hot Encoded Weather Conditions: Clear, Clouds, Drizzle, Fog, Haze, Mist, Rain, Smoke, Snow, Thunderstorm

---

5. Methodology:

1. **Data Preprocessing:**
    - Handled missing values and duplicates
    - Converted datetime into separate features
    - One-hot encoded weather conditions

2. **Model Training:**
    - Split data into training and testing sets
    - Used RandomForestRegressor (from Scikit-learn)
    - Evaluated with Mean Squared Error (MSE)
3. **Model Deployment:**
    - Saved trained model using joblib
    - Developed a Flask-based web app
    - Created a user-friendly form to input features
    - Connected form input to prediction logic

---

6. Application Flow:

- User accesses the form via the homepage (index.html)
- Enters inputs like temperature, rain, snow, hour, date, and weather conditions
- Submits form → Triggers Flask POST request
- Model processes the inputs and returns a predicted traffic volume
- Output is rendered back on the same page

---

7. Challenges Faced:

- Handling mixed data types (datetime, float, bool)
- Managing feature mismatch between training and prediction
- Ensuring form inputs are accurately mapped to model features

---

8. Output Example:

- Inputs:
    - Temperature: 12.5
    - Rain: 0.0
    - Snow: 0.0
    - Hour: 8
    - Month: 5
    - Day: 23
    - Weekday: 4

    - is_weekend: 0
    - is_holiday: 0
    - Weather: Clouds
- Output: Predicted Traffic Volume = 3526 vehicles

---

9. Future Enhancements:

- Add maps and visualizations
- Integrate live weather data APIs
- Include peak-hour logic and congestion alerts
- Export predictions to Excel/CSV

---

10. Conclusion:
The TrafficTelligence project successfully demonstrates the integration of machine learning with web development to build a practical, user-friendly system for predicting traffic volume. It can serve as a foundation for intelligent transport systems and real-time traffic monitoring applications.