Real-Time Traffic Analysis: Comprehensive Report

1. Master Doc/Slides and Learnings

Overview:
This project aimed to analyze real-time traffic conditions by integrating weather data with traffic data for meaningful insights and predictions.

Learnings:

- **Data Integration:**
    - Combined the US Accident Dataset (7.7M records) and OpenWeatherMap API data (~60 KB) for enriched analysis.
    - Learned how external APIs enhance dataset features.
- **Structured Pipelines:**
    - Implemented pipelines for preprocessing, feature engineering, and model evaluation.
- **Visualization Techniques:**
    - Used tools like Seaborn and Matplotlib to visualize class separability and target predictability.

API Challenges:

- Encountered rate limits and data consistency issues with OpenWeather API, requiring fallback strategies.

Slides Key Points:

Objectives:

- Understand data preprocessing, feature engineering, and model-building techniques.
- Develop actionable insights for traffic optimization using weather and traffic conditions

Visualizations :

- Pair plots, correlation matrices, and classification reports highlighted data patterns and model performance.

2. Data processing and future engineering:

Preprocessing Pipeline:

- **Handling Missing Values:**
  - Applied imputation for numerical features and mode substitution for categorical ones.
- **Normalization and Scaling:**
  - Used Min-Max Scaling for numerical features like temperature and distance.
- **Encoding Categorical Features:**
  - OneHotEncoding for nominal variables like Weather_Condition**.**
  - LabelEncoding for ordinal variables like Sunrise_Sunset.
- **Feature Engineering**:

  Derived Features:

- Weather_Severity_Index: Combined visibility and precipitation to create a composite weather impact score.
- Traffic_Intensity: Merged traffic patterns with time of day for congestion prediction.

  Feature Selection:

- Correlation analysis and mutual information metrics identified top predictors.

  Class Separability:

  Visual Insights:

- Scatter plots and pair plots revealed limited separability for minority classes.
- Principal component analysis (PCA) and t-SNE visualizations improved understanding of class clusters

3. Model building and evaluations:

Model Selection:

- **Random Forest Classifier:**
  - Chosen for its robustness and interpretability.
  - Baseline accuracy: 78%, skewed by class imbalance.
- **Gradient Boosting**:

    - Enhanced performance for minority classes, improving F1-scores for underrepresented severity levels.

Evaluation Metrics:

- **Classification Report**:
    - Class 2 (dominant): Precision: **0.85**, Recall: **0.82**, F1-score: **0.83**.
    - Minority classes (1, 3, 4): Poorer performance; Class 1 F1-score: **0.2**
- **Insights :**
- Severe class imbalance affected overall accuracy
- Strong predictors included weather-related features like precipitation and visibility.

4. Model interpretability and tuning:

Tuning Process:

- **Grid Search:**
    - Explored hyperparameters like n_estimators, max_depth, and min_samples_split.
    - Optimal configuration: n_estimators=150, max_depth=12.
- **Class Weight Adjustment:**
    - Addressed imbalance by weighting minority classes higher during model training.

Interpretability:

- **Feature Importance:**
    - Weather_Severity_Index, Traffic_Intensity, and Pressure_Temperature_Diff emerged as top predictors**.**
- **SHAP Values:**
    - Highlighted individual feature contributions, showing weather's significant impact on severity predictions.
- Insights :
- Improving predictions for minority classes remains a challenge

- Future iterations can benefit from exploring alternative models like XGBoost and neural networks

Conclusion

Key Strengths:

- Successfully integrated large-scale historical data with real-time weather data.
- Developed a robust preprocessing pipeline that addressed data quality issues and engineered meaningful features**.**

Challenges:

- Class imbalance significantly impacted model performance for minority severity levels.
- API limitations introduced inconsistencies in weather data.

Future Work:

- Explore advanced ensemble techniques like XGBoost or CatBoost for better handling of imbalanced data.
- Incorporate additional real-time traffic metrics for enhanced predictions.
- Automate the data collection pipeline for seamless integration with APIs