GoTo Data Science Take-Home Assignment: Driver Assignment Summary

1. Objective

Develop a machine learning pipeline to predict driver acceptance of booking requests, enabling optimal driver assignments in a ride-hailing platform.

2. Understanding the Problem

- The core challenge is to accurately predict driver behavior (accept/reject) based on booking details and driver history.
- This involves processing raw data, extracting relevant features, selecting an appropriate model, and optimizing its performance.
- The goal is to create a robust and scalable solution that can be deployed in a real-world ride-hailing system.

3. Data Exploration and Preparation

- Explored the provided dataset (booking_log.csv, participant_log.csv, test_data.csv) to understand its structure, content, and potential issues.
- Identified key variables, data types, missing values, and class distribution.
- Conducted Exploratory Data Analysis (EDA) to discover patterns and relationships in the data.
    - Balanced target variable distribution.
    - Dominance of short trips.
    - Strong correlation between shorter wait times and higher acceptance rates.
    - Influence of driver history and peak hours on acceptance.
- Prepared the data for modeling by cleaning, merging, and splitting it into training and testing sets.

4. Feature Engineering

- Engineered seven features based on domain knowledge and EDA insights: driver_distance, event_hour, historical_completed_bookings, driver_acceptance_rate, wait_time, wait_time_squared, and distance_time_interaction.
- The rationale was to capture factors influencing driver decisions: proximity, time sensitivity, past behavior, and combined effects.
- Handled missing values and normalized numerical features to ensure data quality and model stability.

5. Model Selection and Training

- Evaluated three models: Random Forest, XGBoost, and LightGBM.
- LightGBM was chosen for its superior performance and efficiency.
    - Random Forest was too slow and inaccurate.
    - XGBoost was slower than LightGBM with comparable accuracy.

- Trained a LightGBM model with fixed hyperparameters, balancing accuracy and speed.

6. Addressing Challenges

- Encountered and resolved numerous technical challenges:
  - Import errors, file path issues, Pandas warnings.
  - KeyErrors, TypeErrors, JSON saving errors.
  - Model training and deserialization problems.
  - Performance issues and formatting errors.
- These were addressed through debugging, code corrections, and configuration adjustments.

7. Results and Conclusion

- The LightGBM model achieved 76-80% accuracy.
- Generated results.csv with driver assignments and metrics.json with evaluation metrics.
- The pipeline effectively predicts driver acceptance and can be used for optimal driver assignment.
- Future work may include additional features, ensemble models, hyperparameter tuning, and real-time deployment considerations.