**FUTURE ENHANCEMENT**

### 1. Explore Additional Machine Learning Algorithms

* Implement more sophisticated algorithms such as **Random Forest**, **Support Vector Machine (SVM)**, or **Gradient Boosting Machines (GBM)** to improve accuracy, as done in studies like the one in the paper you uploaded. These models could be implemented similarly to your logistic regression pipeline.
* The paper you uploaded also explores **CNN-LSTM deep learning frameworks** for improving delay predictions. These models capture sequential data better, which could be helpful if you have time-series data or patterns​(Flight\_Delay\_Prediction…).

**2. Feature Engineering**

* **Time-based features**: Incorporate features such as the day of the week, month, or time of the day to improve the accuracy of the predictions. Flight delay data often has time-based patterns.
* **Weather Data**: Integrate weather information, which is often a strong predictor of delays, as suggested by various studies​(Flight\_Delay\_Prediction…).
* **Handling Dates**: Consider deriving features like the time difference between START\_DATE and END\_DATE, time of day (morning/afternoon/evening), and seasonal indicators.

**3. Hyper parameter Tuning**

* Use techniques like **Grid Search** or **Random Search** to optimize model parameters.
* Alternatively, **Bayesian optimization** could provide more efficient tuning for complex models.

**4. Model Evaluation and Cross-Validation**

* Implement **k-fold cross-validation** to better evaluate your model’s performance and ensure that it generalizes well to unseen data.
* Introduce additional metrics such as **precision**, **recall**, **F1-score**, and **ROC-AUC** to get a more holistic understanding of model performance, especially for imbalanced datasets.

**5. Visualization and Interpretability**

* Add more advanced visualizations such as **ROC Curves**, **Precision-Recall curves**, and **Feature Importance charts**.
* For interpretability, you could integrate methods like **SHAP (SHapley Additive exPlanations)** to explain feature contributions.

**6. Use of Deep Learning Models**

* You could experiment with deep learning models like **Convolutional Neural Networks (CNNs)** or **Recurrent Neural Networks (RNNs)**, particularly **LSTMs**, as mentioned in the uploaded document for handling sequential data​(Flight\_Delay\_Prediction…).

**7. Incorporating External Data Sources**

* Use external APIs (e.g., for real-time weather data) to enhance the dataset and potentially improve model accuracy. This would help with real-time predictions and more dynamic modeling.

**8. Deploying the Model as a Web Service**

* To make the model usable in real-world applications, consider deploying it as a REST API using tools like **Flask**, **Django REST Framework**, or **FastAPI**. This would allow external applications to use the model for predictions in real-time.