Notebook 07: Bus Ridership Prediction

This documentation covers **Notebook 07: Bus Ridership Prediction** (`07_bus_ridership_prediction.ipynb`), the final notebook in the analysis pipeline. This module serves a distinct purpose: moving beyond simply analyzing violations to building a robust prediction engine for ridership, a key factor in maximizing the impact of the **ClearLane Initiative** deployment strategies.

This notebook uses machine learning to supplement the primary enforcement analysis, recognizing that effective deployment must prioritize bus routes where the maximum number of people—including CUNY students—will benefit from reduced congestion.

Role and Context in the Project Narrative

This notebook focuses on applying advanced machine learning techniques, specifically a **Random Forest Regressor**, to predict bus ridership volume. This addresses the utilization component of Datathon Question 1 (Which MTA bus routes are highly utilized by CUNY students?) by forecasting where demand is highest. The prediction output will ultimately be incorporated into the Deployment Priority Score used by the overall optimization engine.

Part 1: Data Preparation and Feature Engineering

The analysis begins by loading **7 million subway ridership records from 2025** to train the prediction model, utilizing the Scikit-learn ecosystem for feature processing and model building.

The core of the preparation involves creating granular features necessary for accurate time series prediction, drawing heavily on temporal and spatial indicators:

- **Temporal Features:** The notebook processes the timestamps to extract predictable cyclical patterns, including `month`, `day`, `hour`, `weekday`, and `weekend flags`. These features capture the repetitive rhythms of commuter and student life, which are crucial for forecasting demand.
- **Spatial Encoding:** To allow the model to distinguish between different geographic areas, it creates **borough dummy variables** for key locations, specifically encoding Manhattan, Queens, Brooklyn, the Bronx, and Staten Island.
- **Model Choice:** A **Random Forest Regressor** is selected for this task. This non-linear model is well-suited for high-dimensional data where complex interactions between time (hour, day) and location (borough) determine the ridership volume.

Part 2: Model Training and Performance Validation

The Random Forest model is trained with careful hyperparameter tuning and evaluated against a held-out test set to determine its genuine predictive capability. The performance is assessed using rigorous regression metrics:

- **R^2 Score:** The model achieved an outstanding **R^2 score of 0.963**. This means the model successfully explains **96.3%** of the variance observed in the ridership data, demonstrating highly accurate predictive power.
- **Error Metrics:** The model’s precision is quantified by its error rates:
  - The **Mean Absolute Error (MAE)** is reported as **2903.57**.
  - The **Root Mean Squared Error (RMSE)** is **7940.28**.
  - The **SMAPE** (Symmetric Mean Absolute Percentage Error) is calculated at **33.63%**.

Part 3: Feature Importance and Visualization

Understanding *why* the model makes its predictions is crucial for building trust and ensuring the predictive logic aligns with operational reality.

- **Key Predictors:** A feature importance analysis is performed, which shows that the **hour of the day** is overwhelmingly the ** most important predictor**, accounting for **37.9 percent importance** in the model’s forecast decisions. This confirms that the time-of-day is the single largest determinant of ridership volume.
- **Visual Validation:** To validate the impressive R^2 score, the notebook generates and saves a visualization called **`actual_vs_predicted_ridership.png`**. This scatter plot displays the model’s predictions against the actual observed ridership values, showing how closely the

forecasted points cluster around the line of perfect prediction, visually confirming the model's strong fit.

Part 4: Final Outputs and Deployment Readiness

This notebook's final steps are dedicated to packaging the resulting predictive intelligence for immediate use by the rest of the project pipeline and the Streamlit dashboard:

1. **Model Persistence:** The final, optimized **Random Forest model** is saved to disk as a pickled file, **`rf_best_model.pkl`**, using the `joblib` library. This allows the complex model to be loaded quickly into the deployment optimizer (implied, though not explicitly shown in the code) or the Streamlit environment without needing to be retrained.
2. **Prediction Function:** A utility function, **`predict_and_show_ridership`**, is defined to handle new data inputs (such as specific month, day, hour, and borough) and generate ridership forecasts. This function is essential for dynamically querying the model during deployment simulations or interactive dashboard use, handling the necessary internal logic, including encoding the borough names correctly before feeding data to the saved model.
3. **Text Export:** The complete documentation, detailing the model's training process, performance metrics, and feature importance findings, is saved as the text file **`results/txt/07_bus_ridership_prediction.txt`**.

In summary, Notebook 07 successfully delivers a high-performance machine learning model capable of accurately predicting ridership volume, providing the crucial predictive layer needed to guide the **ClearLane Initiative** to maximize its positive impact on public transit effectiveness.