Predicting Food Delivery Time Using Machine Learning

Presented By Yodha Pranata

Yodha Pranata

LinkedIn:
linkedin.com/in/yodha-pranata

Introduction

Work Experiences

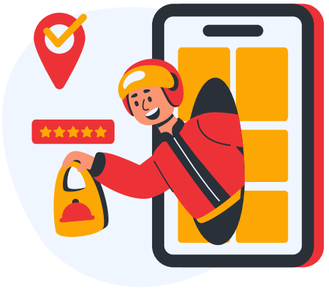
- Critical Care Nurse
  Public Hospital (2019 - 2024)
- Career Center Coordinator
  University of Kepanjen (2018 - 2019)

Education

Bachelor of Data Science
Universitas Terbuka (2024 - Present)

Master of Critical Care
Universitas Padjadjaran (2024 - Present)

Bachelor of Nursing Science & Nursing Profession
Universitas Brawijaya (2012 - 2017)



Project Background

Reliable delivery time and **accurate ETAs** are critical drivers of customer experience in online food delivery. **Late deliveries** can **reduce** perceived **service quality** and **customer trust,** while accurate ETAs help set expectations and support better operational decisions such as dispatching, routing, and staffing.

Predicting delivery time is challenging because delivery performance depends on multiple dynamic factors that change throughout the day. ***Restaurant preparation delays***, ***traffic congestion***, and ***weather disruptions*** can **cause high** variability in **delivery duration**.

Project Background

Inaccurate ETAs can also increase operational inefficiency, including courier idle time, order reassignments, and higher customer support workload. **Data-driven modeling** is widely used in last-mile operations to estimate **delivery duration more reliably** and to support decision-making under uncertainty. This approach enables organizations to quantify key drivers of delays and improve operational planning.

Problem Statement

How accurately can **food delivery time** be predicted, and which operational **factors influence** delivery time the most?

Objective

To develop and **evaluate** an **ETA prediction model** that improves delivery time estimation accuracy and **provides actionable insights to support operational decisions**

Data Understanding

1. **Data source:** Public dataset from **Kaggle**
2. **Dataset size:** **1000 rows** and **9 columns**

- **Order_ID:** Unique identifier for each order.
- **Distance_km:** The delivery distance in kilometers.
- **Weather:** Weather conditions during the delivery, including Clear, Rainy, Snowy, Foggy, and Windy.
- **Traffic_Level:** Traffic conditions categorized as Low, Medium, or High.
- **Time_of_Day:** The time when the delivery took place, categorized as Morning, Afternoon, Evening, or Night.
- **Vehicle_Type:** Type of vehicle used for delivery, including Bike, Scooter, and Car.
- **Preparation_Time_min:** The time required to prepare the order, measured in minutes.
- **Courier_Experience_yrs:** Experience of the courier in years.
- **Delivery_Time_min:** The total delivery time in minutes (target variable).

Data Preprocessing
for EDA

1 Handling Duplicate

No Duplicate Data Found

2 Handling Missing Value

- **Categoric cols missing** → modus imputation
- **Numeric cols missing** → KNNImputer (n_neighbors=5, weights="distance")

3 Handling Outlier

No Outlier Found

Data Preprocessing
for Machine Learning

SPLIT TRAIN & TEST

1 Handling Duplicate

No Duplicate Data Found

2 Handling Missing Value

- **Categoric cols missing** → modus imputation
- **Numeric cols missing** → KNNImputer (n_neighbors=5, weights="distance")

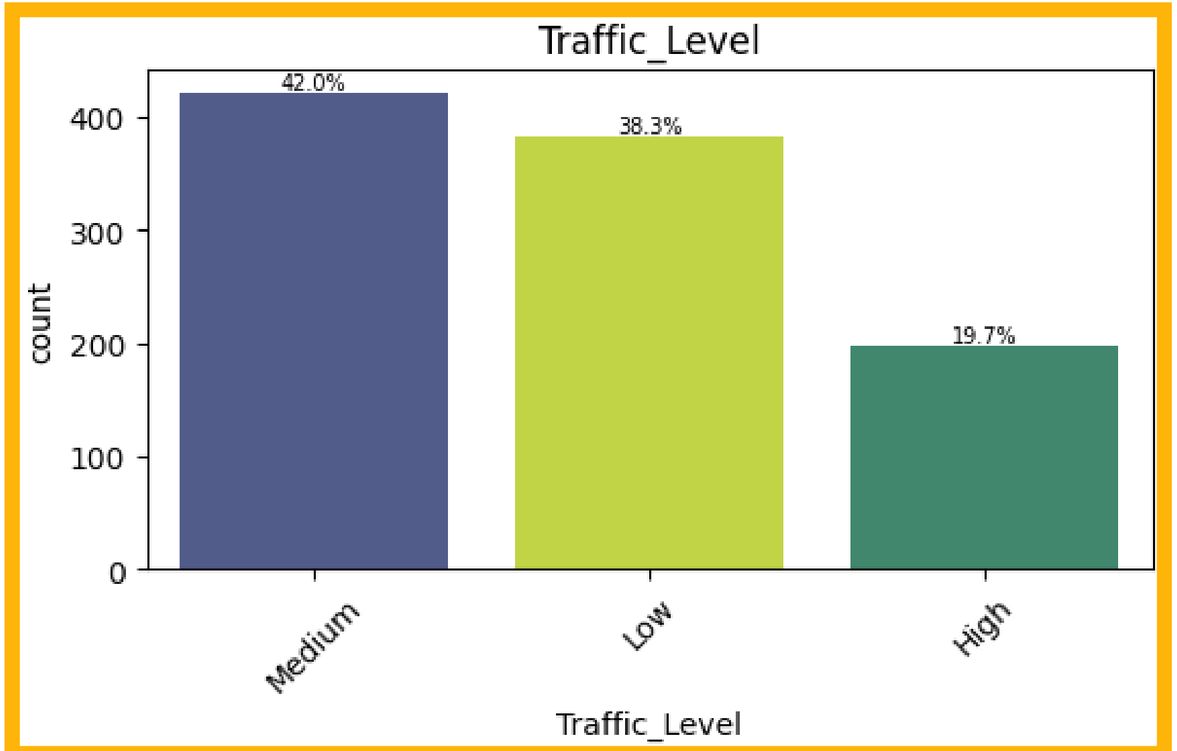
3 Handling Outlier

No Outlier Found

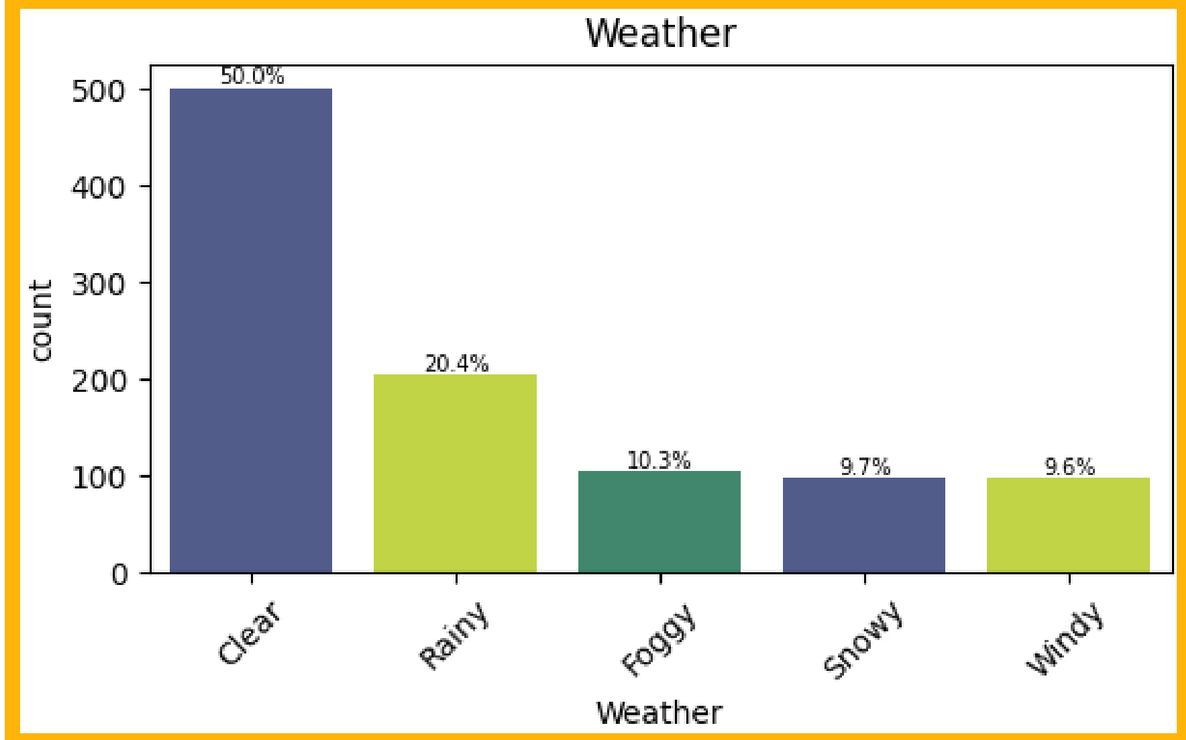
4 Feature Scalling

- **OHE:** 'Traffic_Level', 'Weather'
- **Standard Scaller:** 'Distance_km', 'Preparation_Time_min', 'Courier_Experience_yrs'

Exploratory data analysis

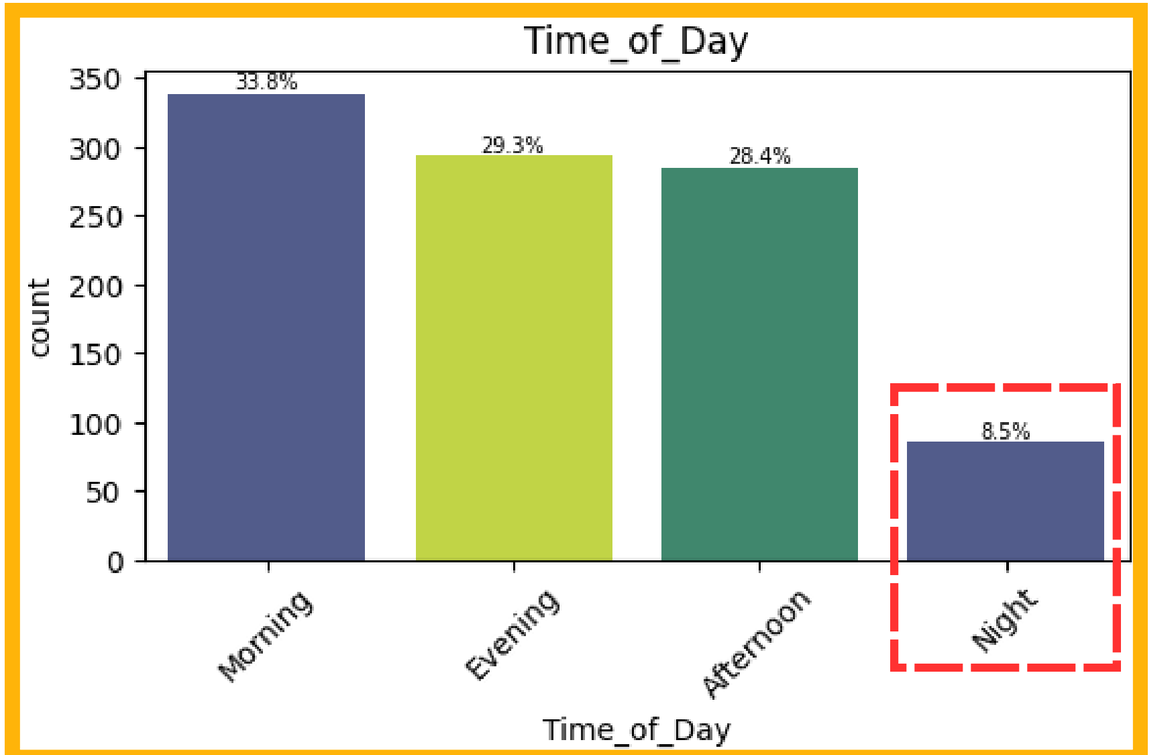


Most orders happen when traffic is Medium (42.0%) or Low (38.3%). High traffic (19.7%) is less common, so the data **mostly represents normal road** conditions.

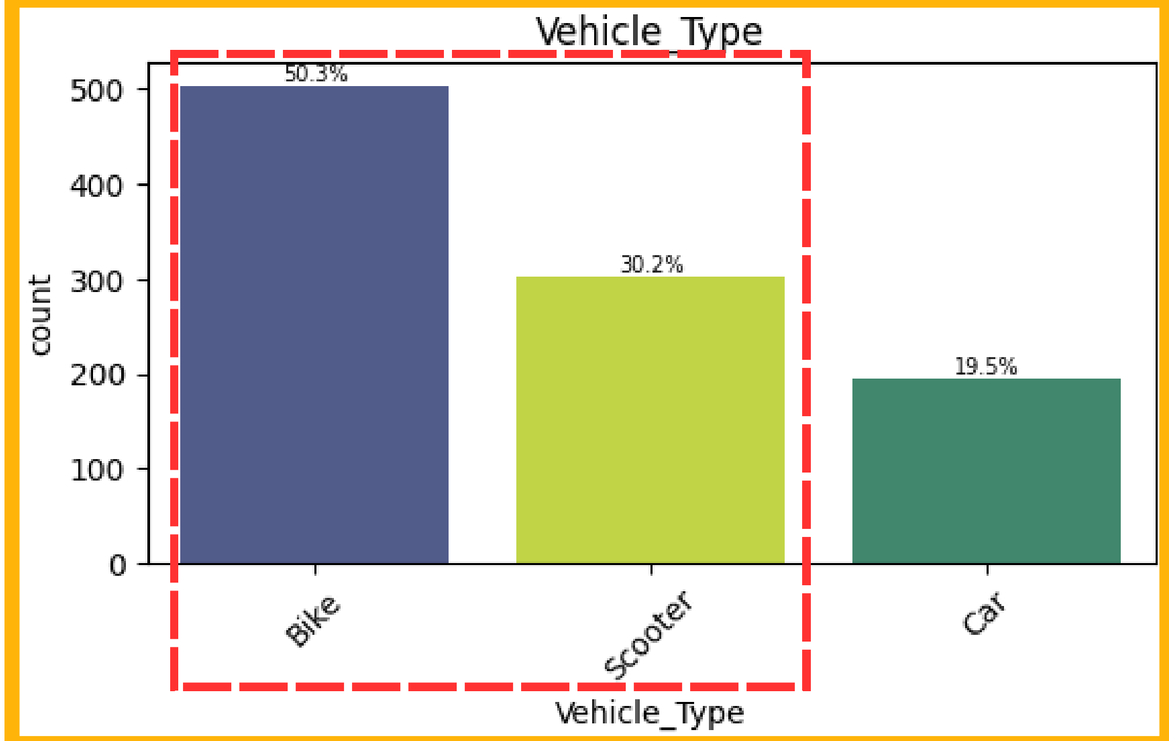


Conditions are dominated by **Clear weather (50.0%),** followed by Rainy (20.4%). Foggy (10.3%), Snowy (9.7%), and Windy (9.6%) appear relatively evenly but with smaller shares, meaning **extreme-weather patterns are present but less common.**

Exploratory data analysis

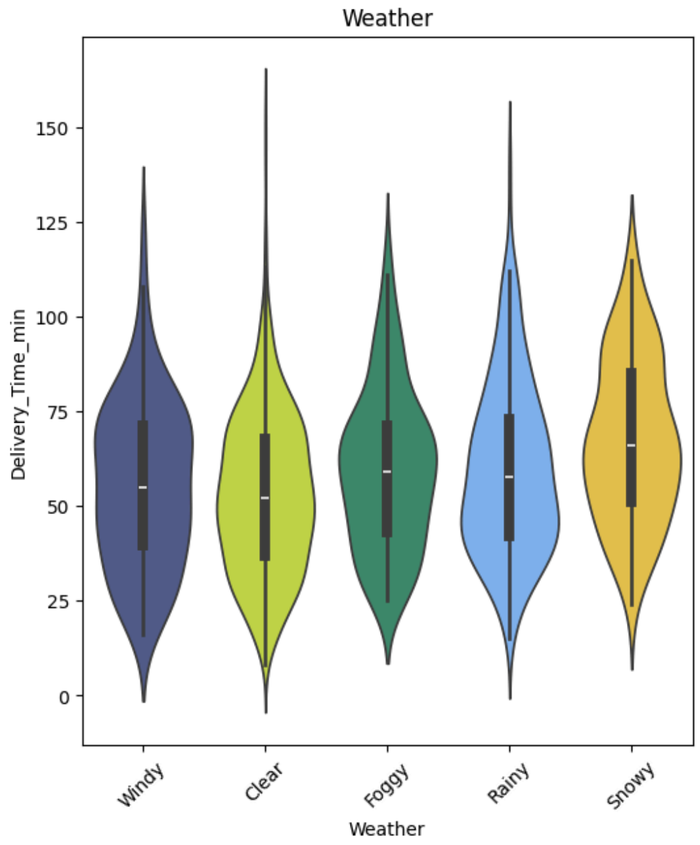


Deliveries are **concentrated in Morning** (33.8%), Evening (29.3%), and Afternoon (28.4%), with Night (8.5%) underrepresented. **Model insights for night-time** deliveries may be **less stable due to the smaller sample.**

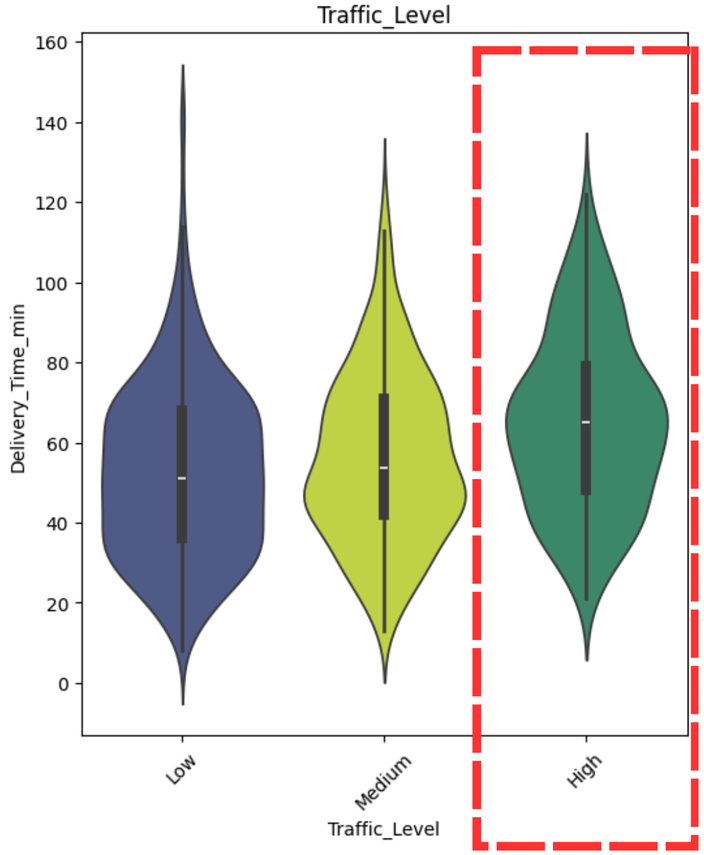


Bikes (50.3%) are used the most, then Scooters (30.2%), and Cars (19.5%). This means the dataset mainly reflects t**wo-wheel delivery patterns.**

Exploratory data analysis

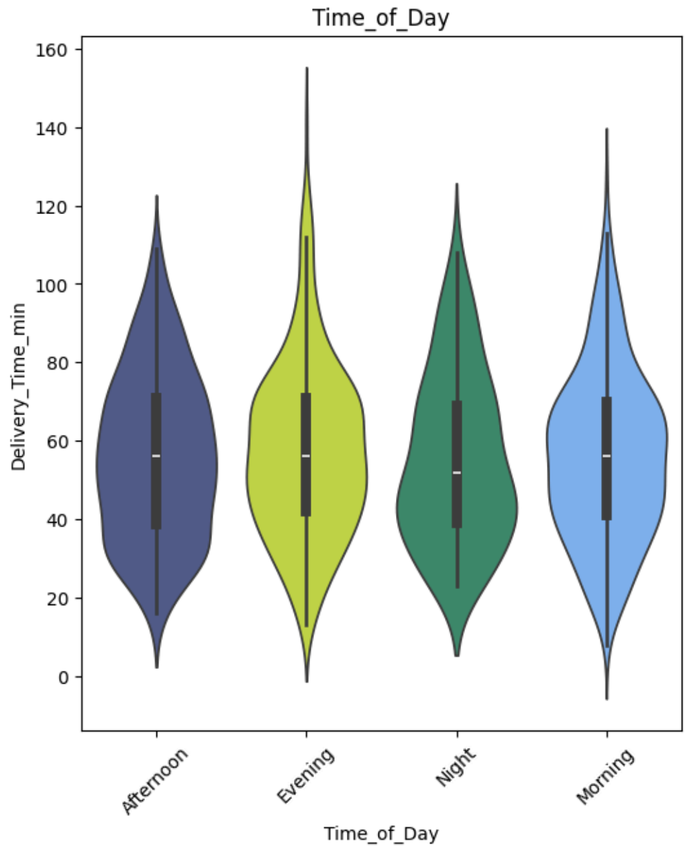


Delivery times vary across weather categories, with some conditions showing higher medians and broader distributions than others. **The Kruskal–Wallis test is significant (p < 0.001),** so weather is associated with meaningful differences in delivery time.

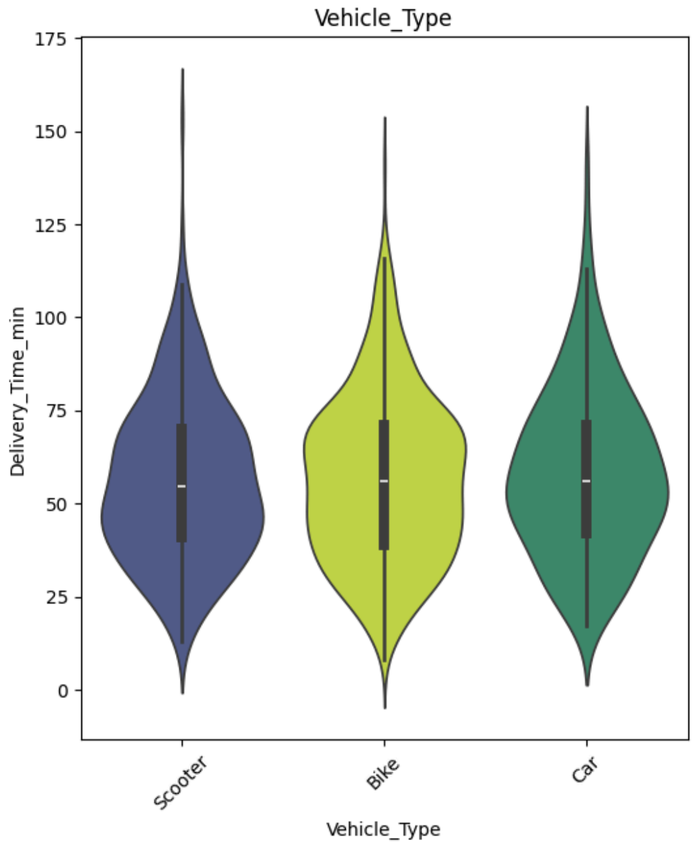


The **High traffic** group shows a **higher median** and a wider spread, indicating longer and more variable delivery times. This matches the **Kruskal–Wallis result (p < 0.001)**, meaning delivery times differ significantly across traffic levels.

Exploratory data analysis

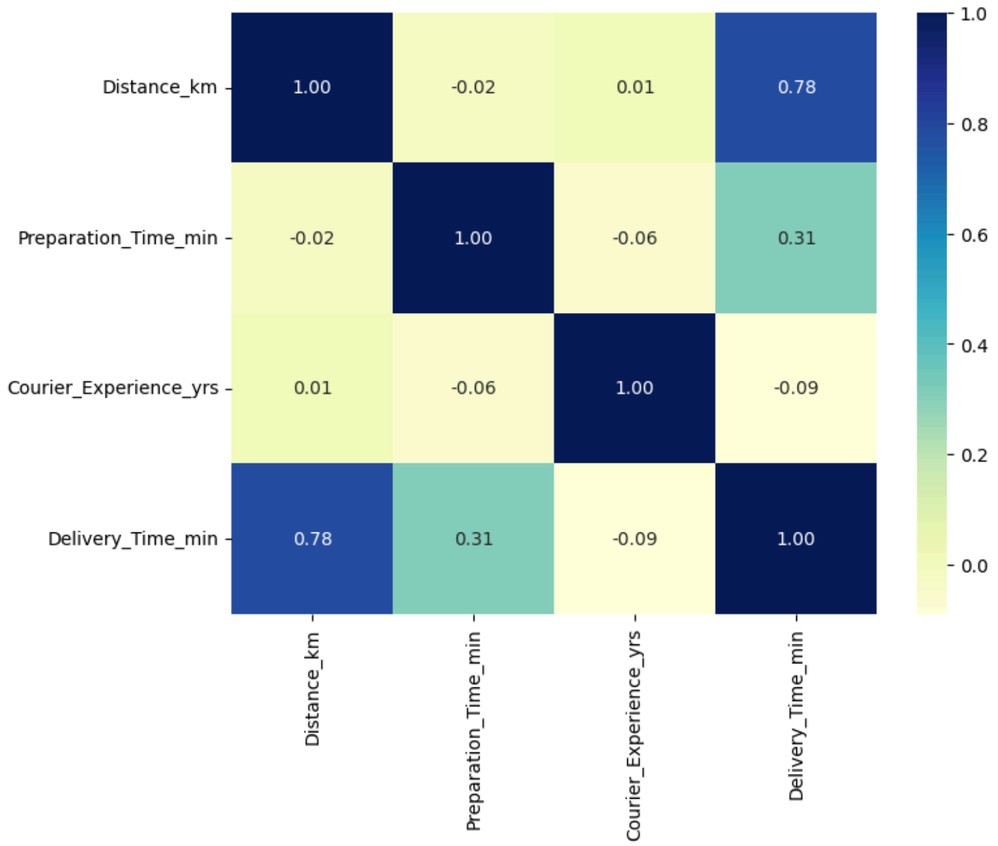


The distributions for Morning, Afternoon, Evening, and Night overlap heavily, and medians are quite similar. **This aligns with the non-significant** Kruskal–Wallis result (p = 0.816), suggesting time of day does not strongly separate delivery times in this dataset.



Scooter, Bike, and Car show very similar medians and overlapping spreads. The Kruskal–Wallis test is not significant (p = 0.611), indicating vehicle type does not create clear differences in delivery time on its own.

Multicollinearity & Feature Selection



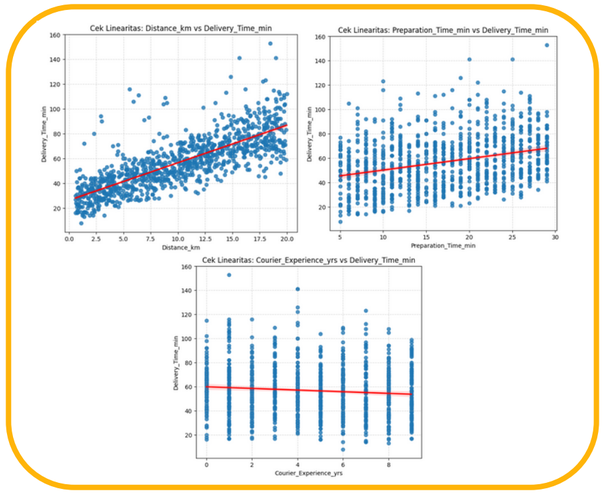
- **No multicollinearity in numeric features**: correlations are near zero and VIF < 5 (Distance 3.10; Preparation 1.35; Experience 1.02).
- **Keep Features:** ***Weather*** and ***Traffic_Level*** (Kruskal–Wallis $p < 0.001$).
- **Drop Features:** ***Time_of_Day*** ($p = 0.816$) and ***Vehicle_Type*** ($p = 0.611$) because they show no meaningful relationship with delivery time (screening threshold $p < 0.25$).

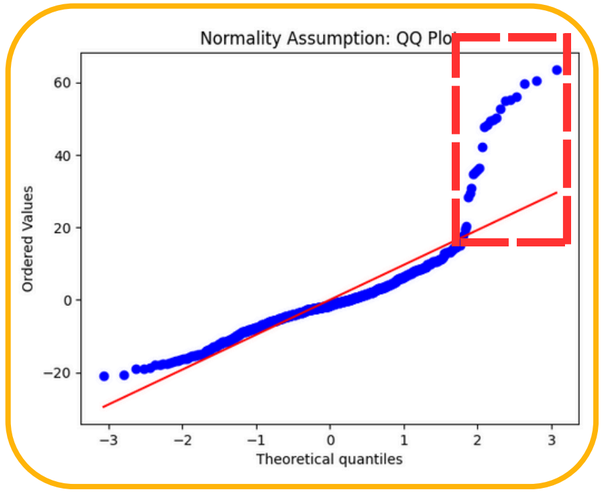
	Features	VIF
1	Distance_km	3.102276
2	Preparation_Time_min	1.345937
3	Courier_Experience_yrs	1.024961
4	Delivery_Time_min	3.442897

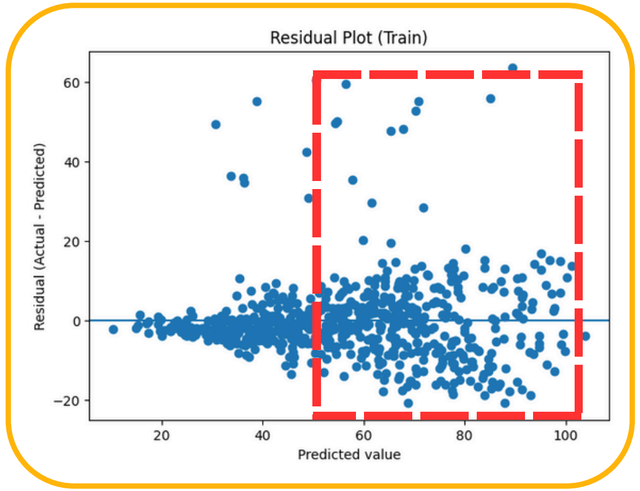
Model Performance
after Tuning

Model	R^2 Train	R^2 Test	RMSE (min)	MAE (min)	Note
Linear Regression	0,767	**0,821**	**8,96**	**6,09**	Best overall (lowest RMSE & MAE)
Ridge	0,766	0,819	9,02	6,20	Stable, but slightly worse than Linear Regression
Lasso	0,765	0,818	9,02	6,15	Similar to Ridge; slightly better MAE than Ridge
XGBRegressor	**0,878**	0,811	9,21	6,54	Train > test → mild overfitting; worse overall

Linear Regression
Assumption Checks







	Features	VIF
1	Distance_km	3.102276
2	Preparation_Time_min	1.345937
3	Courier_Experience_yrs	1.024961
4	Delivery_Time_min	3.442897

Linearity:

Linearity is reasonably satisfied for *Distance_km* and *Preparation_Time_min* because both show a clear upward trend, while *Courier_Experience_yrs* **shows a weak**, slightly negative linear trend that is **still acceptable for a linear model.**

Normality (Q–Q Plot):

Residuals follow the line in the center but deviate in the upper tail, indicating **non-normal residuals** due to outliers/heavy tails.

Homoscedasticity (Residual Plot):

Residual spread increases at higher predicted values, **showing heteroscedasticity** (error variance is not constant).

Multicollinearity (VIF/Correlation):

Earlier VIF results are below common thresholds (e.g., VIF < 5), so **multicollinearity is not a concern.**

Linear Regression remains appropriate for prediction as a strong baseline based on its test performance.

Caution is needed for strict inference and for very long delivery times, where errors tend to be larger.

Segment-Based Model Evaluation (IQR)

Why segment the evaluation?

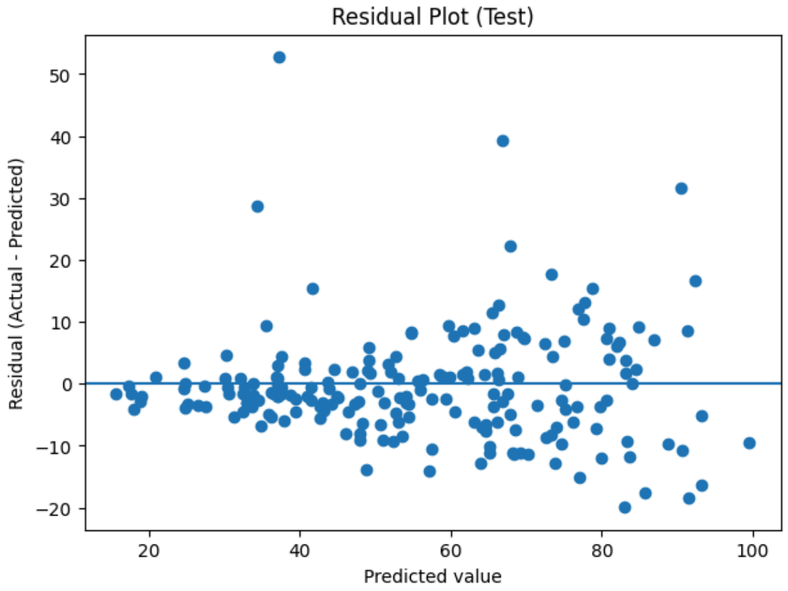
Residual diagnostics indicate heteroscedasticity, meaning prediction errors increase as delivery time becomes longer. Segmenting delivery time using the IQR range helps show where the model performs well and where it struggles.

Linear Regression Performance by Delivery Time Segment

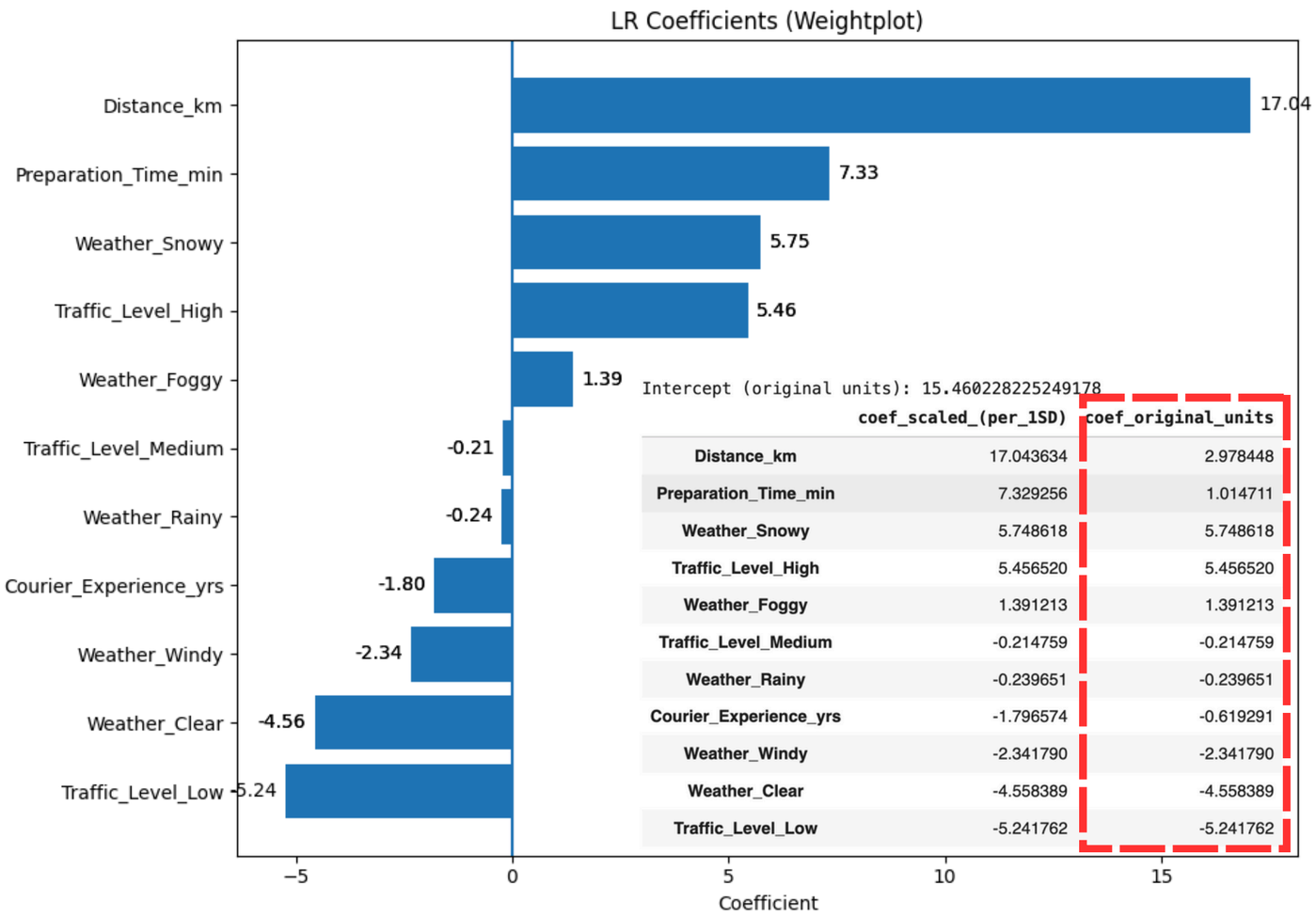
- **≤ 41 minutes** (n = 59): **RMSE 4.075, MAE 3.202 → strong accuracy for short deliveries**
- **41–71 minutes** (n = 94): **RMSE 7.571, MAE 5.656 → stable performance for typical deliveries**
- **> 71 minutes** (n = 47): RMSE 14.365, MAE 10.591 → highest error; long deliveries are harder to predict due to higher variability and outliers

Key Insight

The model is reliable for *short* and *normal* deliveries, but prediction quality drops for long deliveries (>71 minutes). Operational actions should focus on managing long-delivery cases and improving data/features for extreme delays.



Feature Importance



Intercept (original units): 15.460228225249178

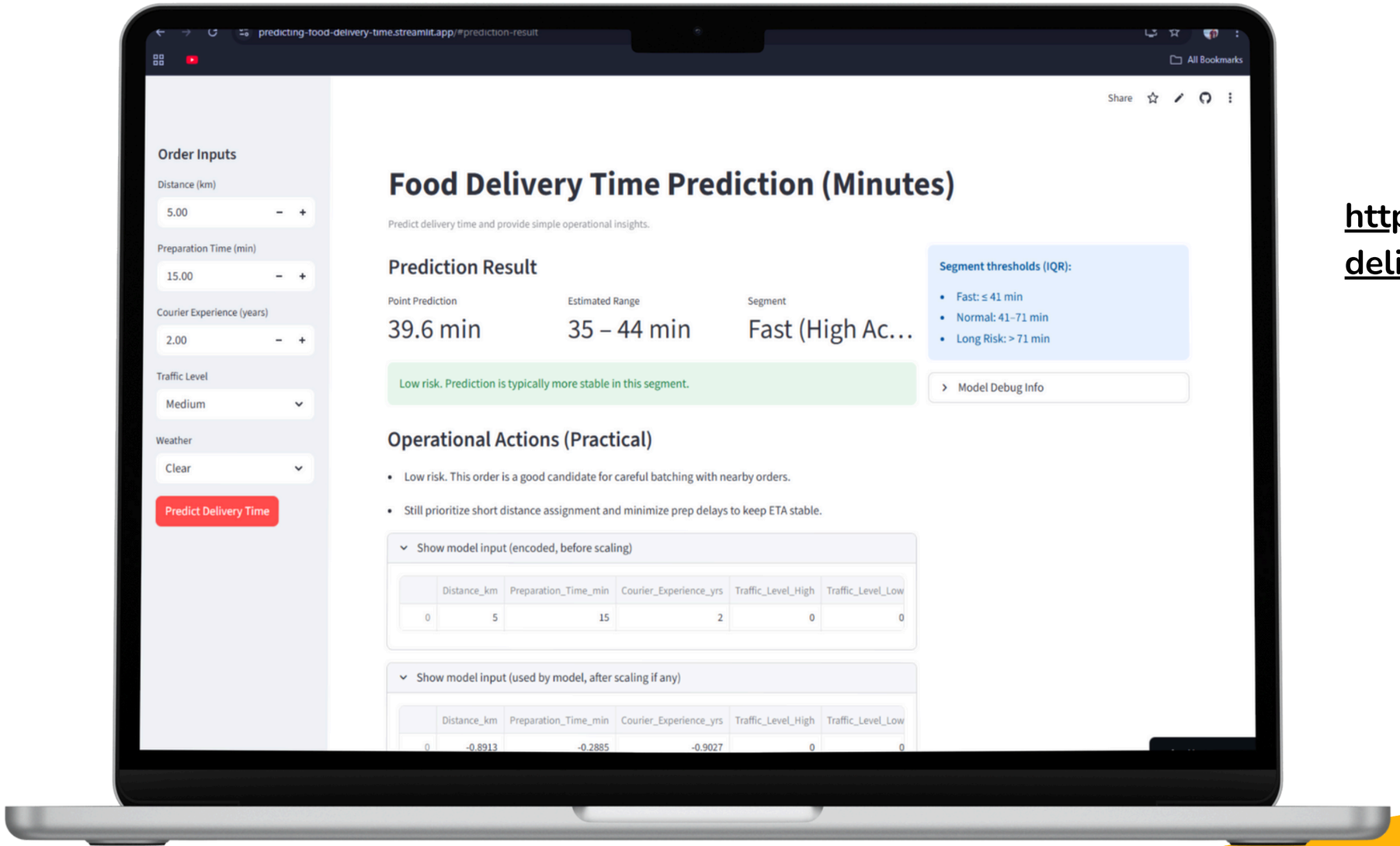
	coef_scaled_(per_1SD)	coef_original_units
Distance_km	17.043634	2.978448
Preparation_Time_min	7.329256	1.014711
Weather_Snowy	5.748618	5.748618
Traffic_Level_High	5.456520	5.456520
Weather_Foggy	1.391213	1.391213
Traffic_Level_Medium	-0.214759	-0.214759
Weather_Rainy	-0.239651	-0.239651
Courier_Experience_yrs	-1.796574	-0.619291
Weather_Windy	-2.341790	-2.341790
Weather_Clear	-4.558389	-4.558389
Traffic_Level_Low	-5.241762	-5.241762

- **Baseline:** The model's base prediction is about 15.5 minutes (intercept), before adding effects from distance, prep time, traffic, and weather.
- **Distance is the main driver:** Each +1 km increases delivery time by about +3.0 minutes.
- **Restaurant prep time matters:** Each +1 minute of preparation adds about +1.0 minute to delivery time.
- **Courier experience helps slightly:** Each additional +1 year of experience reduces delivery time by about 0.6 minutes.
- **Operational conditions** add moderate impact:
  - High traffic: +5.5 minutes
  - Snowy weather: +5.7 minutes
  - Low traffic: −5.2 minutes
  - Clear weather: −4.6 minutes
- **Key Insight: Focus first on shorter distance** assignment and **reducing prep delays**. Traffic/weather are secondary but still meaningful.

Business Recommendation

- **Reduce delivery distance (top priority). Assign orders to the nearest couriers** and **limit delivery zones** to shorten travel time and improve delivery time accuracy.
- **Control restaurant preparation time. Track slow vs fast restaurants**, reward consistent fast preparation, and dispatch couriers only when orders are close to ready to reduce waiting time.
- **Special handling for long deliveries (>71 minutes). Flag high-risk orders early**, reassign to closer couriers when needed, avoid giving extra orders to the same courier for these cases, and send more frequent delivery time updates to customers.
- **Rules for heavy traffic and bad weather. Increase courier supply** in affected areas and time windows, adjust predicted delivery time automatically, and notify customers earlier when delays are likely.
- **Add operational data to improve extreme-delay prediction. Collect signals** such as restaurant queue status, **"ready for pickup" status**, number of orders carried by the courier, and route-based traffic **to reduce errors for very long deliveries.**

Streamlit App



https://predicting-food-delivery-time.streamlit.app/

Thank You

LinkedIn https://www.linkedin.com/in/yodha-pranata/

GitHub https://github.com/yodhata/Predicting-Food-Delivery-Time