Predictive Analytics for Optimizing "Time-to-Export" Lead Times

Kenya's Horticultural Supply Chain.

1/2026

> *"Every year, Kenya's vital horticulture industry loses billions in revenue due to unpredictable shipping delays that spoil perishable exports like flowers and fresh produce.*
>
> *Our solution is a predictive analytics platform that uses machine learning to forecast 'Time-to-Export' lead times. By analyzing factors like port congestion, document readiness, and shipment details, we give exporters and logistics teams an early warning system to anticipate delays before they happen.*
>
> *This means fewer missed market windows, reduced spoilage, and optimized logistics planning—turning guesswork into reliable, data-driven insight to protect profits and strengthen one of Kenya's most critical export sectors."*

Business Understanding

Problem Statement

Kenya's horticulture sector faces chronic and worsening logistics challenges including:

- Severe airfreight capacity shortages and skyrocketing costs at JKIA
- Frequent customs clearance delays and electronic system failures
- Port congestion and slow turnaround at Mombasa
- External disruptions (weather events, strikes, global shipping crises, ad hoc levies

Main Objective

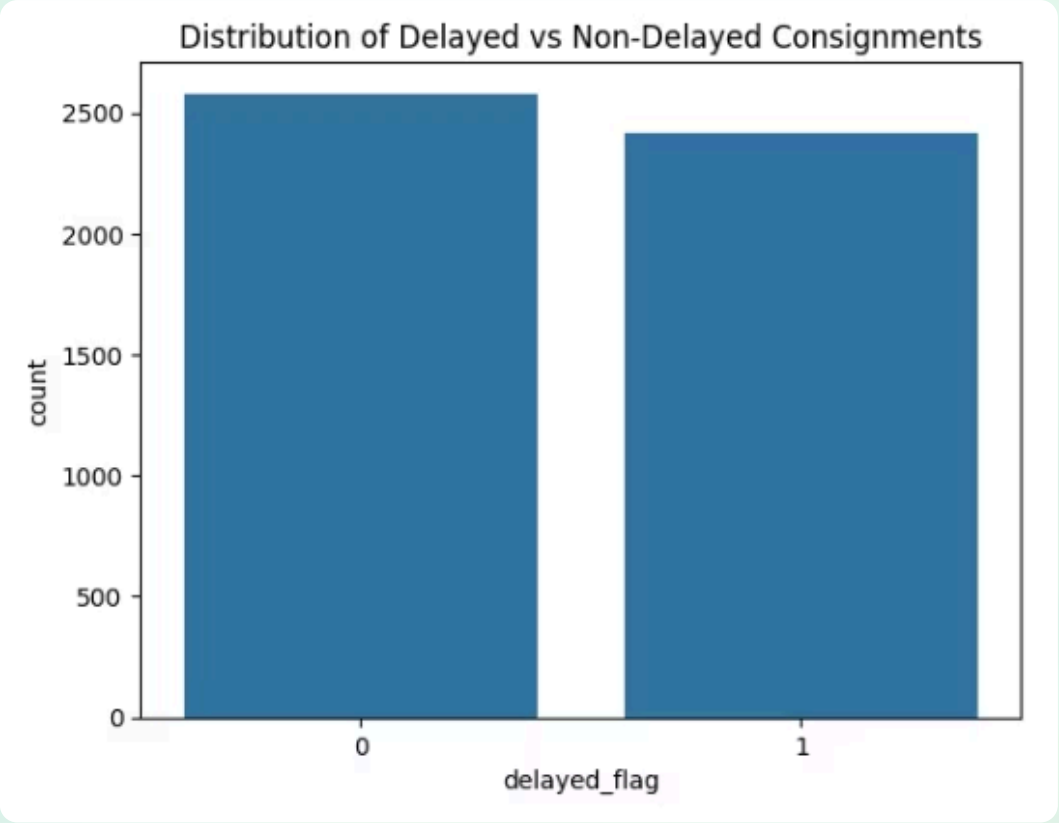
To develop a robust, accurate machine learning regression model that predicts total export lead time (in hours or days) for Kenyan horticultural shipments, enabling proactive planning and risk mitigation.

Stakeholders

- International freight forwarders
- Customs & port operators
- Exporters of perishable commodities
- Supply chain analytics teams

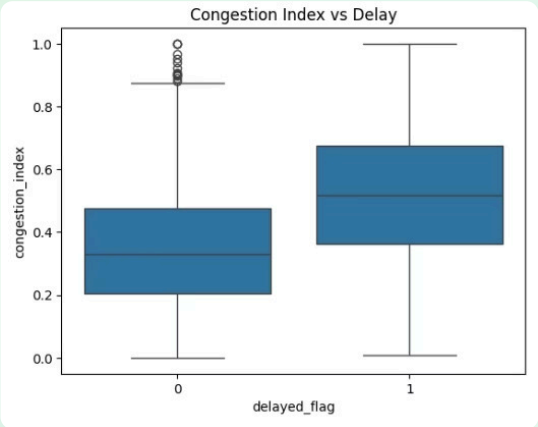
Data Understanding

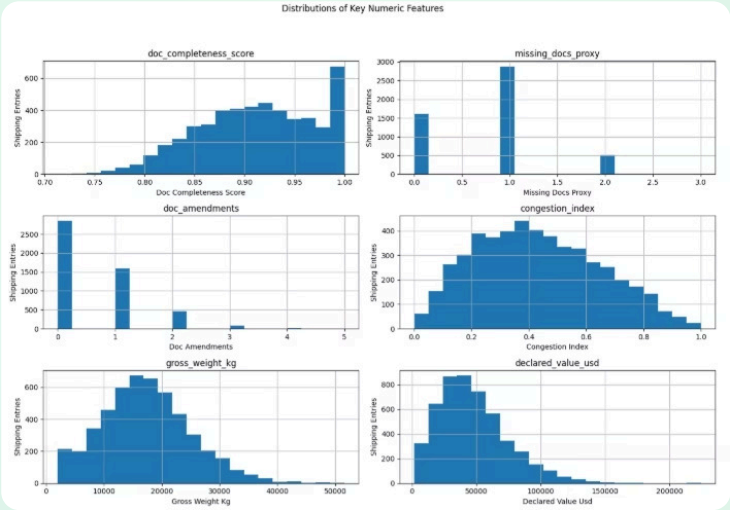
- The dataset contains exactly 5,000 total consignments .
- 27 Features/Columns
- 0 Missing Values
- 5 Origin Countries
- No imputation, dropping, or filling of missing values is performed (none needed)



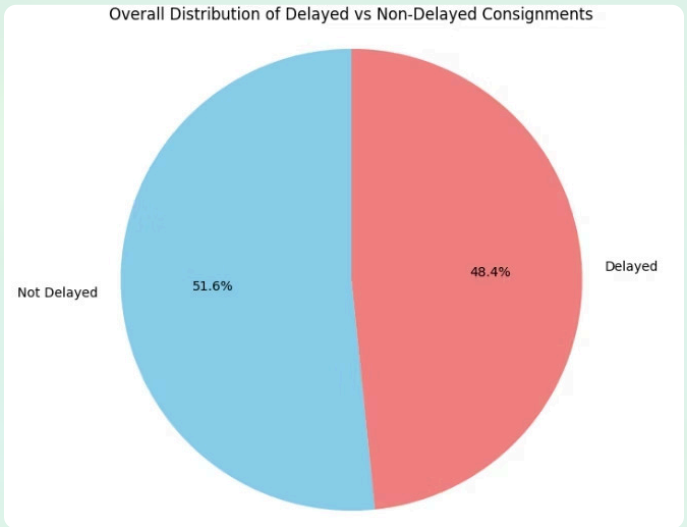
Exploratory Data Analysis (EDA)

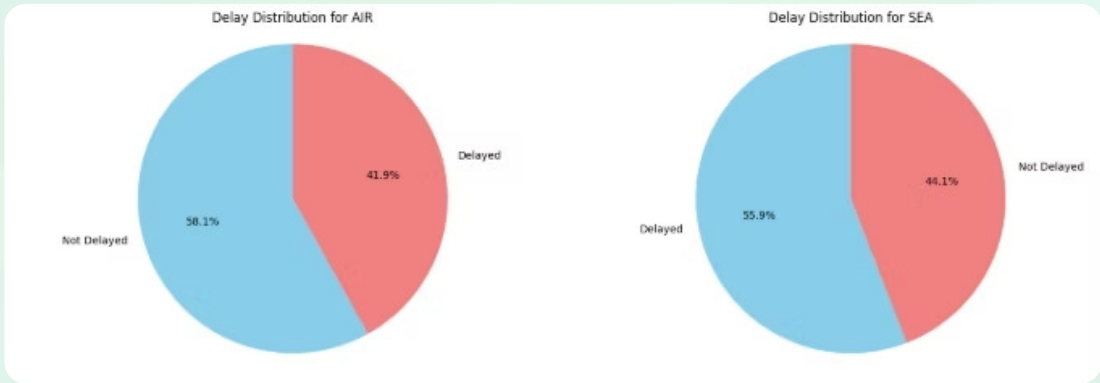
- **Numeric Features:** Most shipments have high `doc_completeness_score` and low `doc_amendments`, while `congestion_index` shows variability.
- **Congestion vs Delay:** Higher congestion is associated with delays.





- **Overall Delay:** ~35% of shipments were delayed, ~65% on time.
- **Shipment Mode vs Delay:** Certain modes experience more delays; e.g., Air shipments delayed ~20%, Sea shipments delayed ~50%.





Data Preparation & Modeling Approach

Preprocessing

- Scaling numeric features
- One-hot encoding categorical variables
- Stratified 80/20 train-test split

Models Evaluated

- Logistic Regression
- Decision Tree
- Random Forest
- XGBoost
- LightGBM

Model	Accuracy	Precision	Recall	F1	ROC_AUC
Logistic Regression	0.813	0.809	0.804	0.806	0.898
XGBoost	0.806	0.813	0.779	0.795	0.890
Random Forest	0.797	0.800	0.775	0.787	0.877
LightGBM	0.795	0.794	0.779	0.786	0.888
Decision Tree	0.777	0.810	0.705	0.754	0.853

- **Logistic Regression:** Best overall performance (highest F1 and ROC-AUC)
- **Tree-based models (XGBoost, Random Forest, LightGBM):** Competitive but slightly lower performance
- **Decision Tree:** Weakest model with lower recall and F1 score

Model Evaluation & Results.

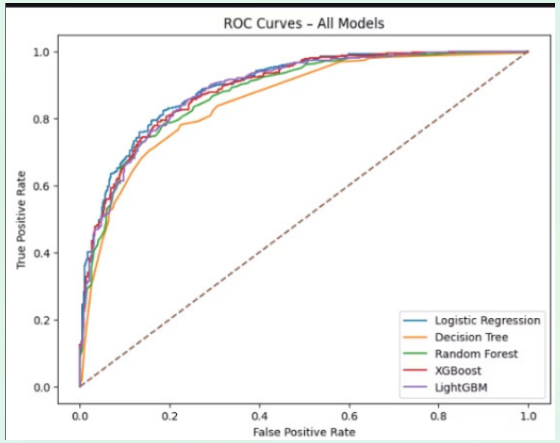
Best Performing Models (ROC-AUC)

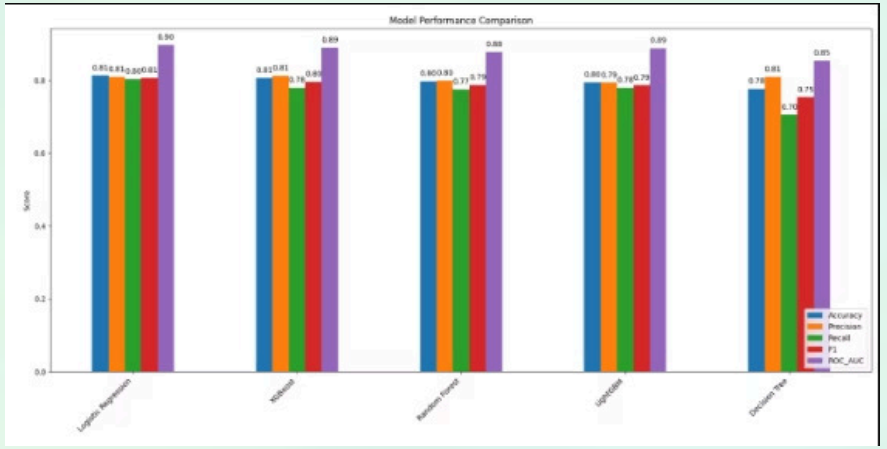
- Logistic Regression: **0.898**
- XGBoost: 0.890
- LightGBM: 0.888

Why Logistic Regression?

- Best F1-score balance
- Handles class imbalance well
- Highly interpretable

Model	ROC_AUC
Logistic Regression	0.898
XGBoost	0.890
LightGBM	0.888
Random Forest	0.877
Decision Tree	0.853





Hyperparameter Tuning

Optimized Parameters

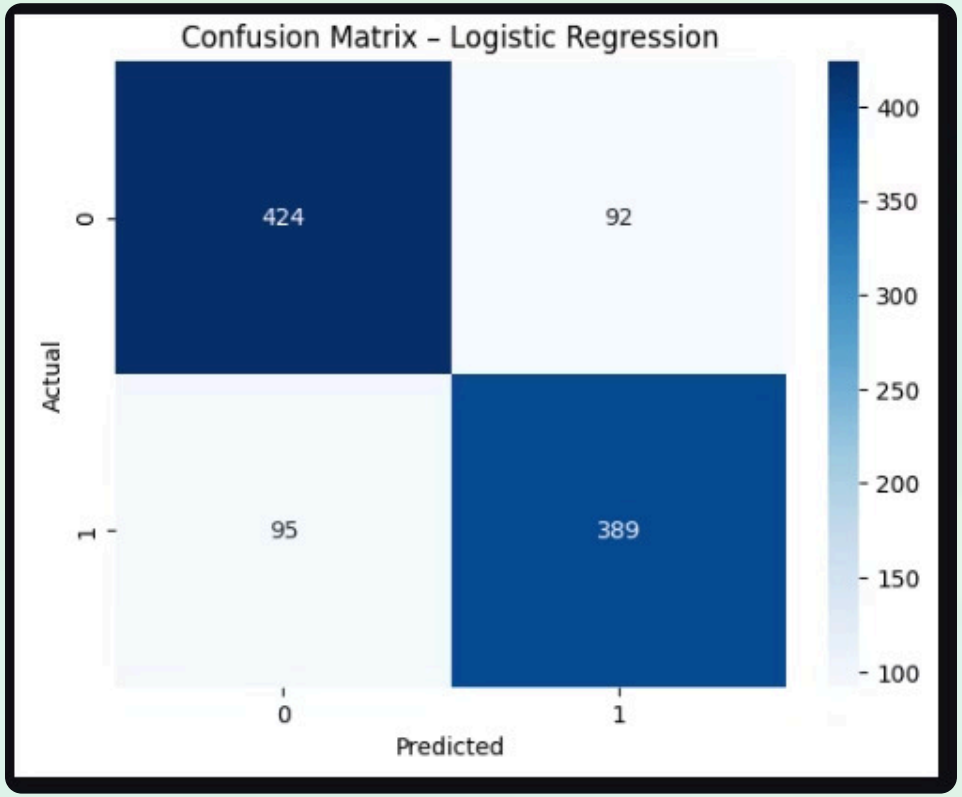
- **n_estimators:** Number of trees
- **max_depth:** Tree depth control
- **min_samples_split/leaf:** Overfitting prevention
- **max_features:** Feature selection strategy

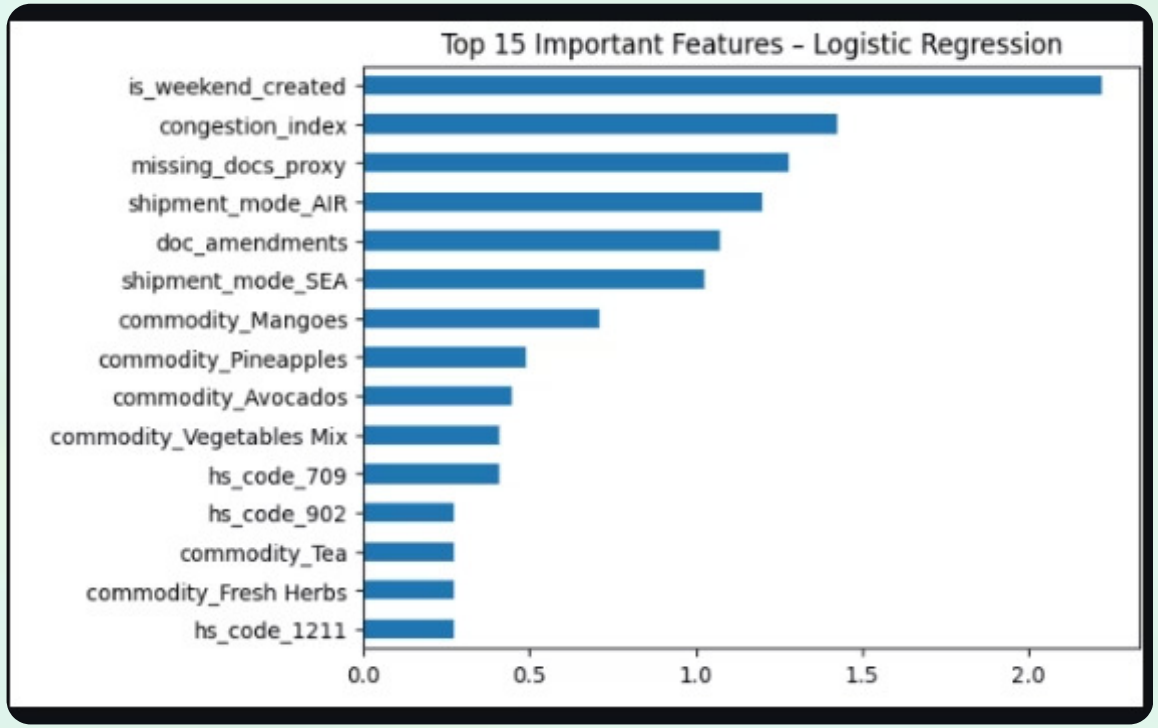
Tuning Balance

- Predictive performance
- Generalization ability
- Computational efficiency

Best Model Diagnostics.

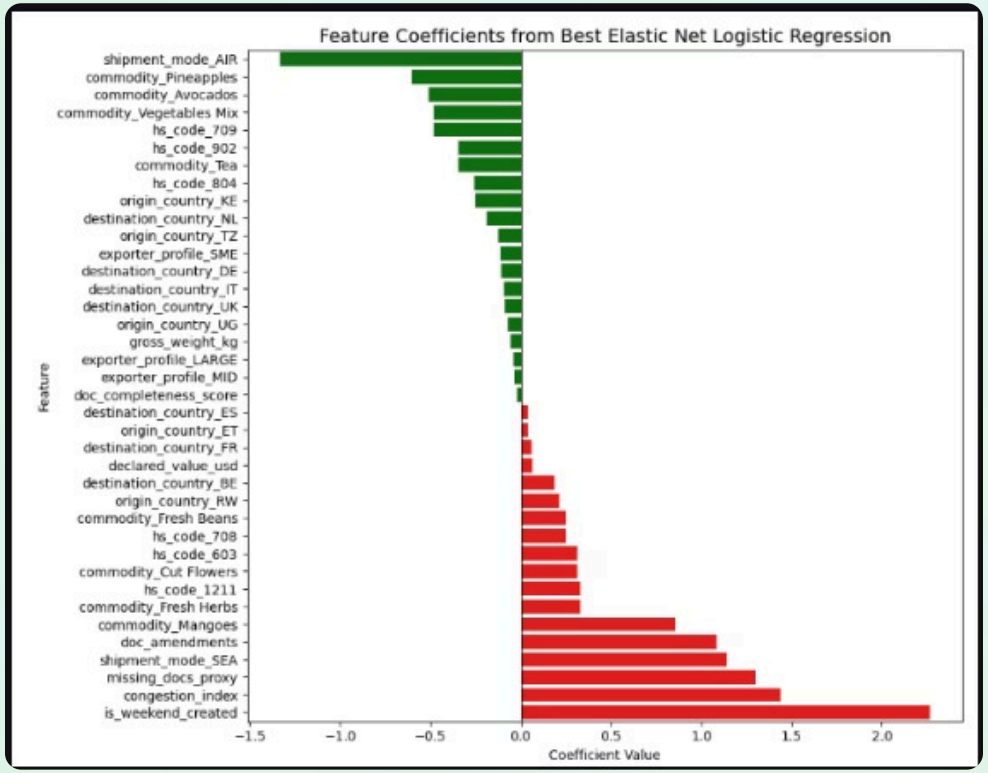
- **Best Model:** Logistic Regression
- **F1-score: 0.81**
- **Diagnostics:** Confusion matrix shows balanced error trade-offs under class imbalance
- **Explainability:** Top 15 features (via coefficients) identify key drivers of shipment delays





Model Iteration: Ridge, Lasso, & Elastic Net Logistic Regression

- Baseline **Logistic Regression** confirmed as best model
- Tested **regularized variants**:
  - Ridge (L2)
  - Lasso (L1)
  - Elastic Net
- Goal: Optimize performance & identify key predictive features



Model	Accuracy	F1	ROC_AUC
Logistic (Unregularized)	0.813	0.806	0.898
Logistic Ridge (L2)	0.813	0.806	0.898
Logistic Lasso (L1)	0.813	0.806	0.898
Logistic Elastic Net	0.813	0.806	0.898

Model Re-Evaluation Using Selected Features:

- Re-evaluate models using only key features from Elastic Net to focus on influential predictors.

Perfomance Using Selected Features

Model	Accuracy	Precision	Recall	F1	ROC_AUC
Decision Tree	0.733	0.727	0.717	0.722	0.733
Random Forest	0.799	0.802	0.777	0.789	0.878
XGBoost	0.788	0.789	0.767	0.778	0.871
LightGBM	0.797	0.801	0.773	0.787	0.888

- Operational Timing & Congestion: Weekend shipments and high congestion drive delays.
- Shipment Mode: AIR reduces risk; SEA increases risk.
- Documentation: Missing or amended documents increase likelihood of delay.
- Commodities & HS Codes: Some commodities and codes slightly increase or reduce risk.

Conclusion & Deployment

- **Best Model:** Elastic Net Logistic Regression – combines L1 feature selection & L2 stability
- **Performance:** Outperforms tree-based models while remaining interpretable
- **Insights:** Coefficients highlight critical risk factors for shipment delays
- **Operational Actions:**
- Monitor high-risk shipments (weekends, congested routes)
- Ensure document completeness
- Allocate resources and improve processes based on model insights
- **Business Value:** Transparent, interpretable model enabling data-driven decisions

Deployment:

Web Application: **https://hortpreddelay.streamlit.app/** *Real-time delay predictions with same preprocessing pipeline*

- **Deployment:** Final Random Forest model launched as a web app for real-time delay prediction
- **Functionality:** Users enter shipment details and receive instant predictions
- **Pipeline:** Same preprocessing steps from training applied, enabling smooth transition from development to practical use

Thank You

Any Questions?