Flight Telemetry Regression Analysis - Final Report

Project: Flight Duration Prediction using Telemetry Data **Author:** Bruno Silva **Date:** 2025-12-02 22:04:37 **Objective:** Develop and evaluate regression models to predict flight duration

---

Environment:

- pandas: 2.3.1
- NumPy: 2.2.6
- scikit-learn: 1.7.2

---

1. INTRODUCTION

1.1 Problem Statement

Flight duration prediction is critical for airline operations, affecting:

- **Schedule Planning:** Accurate duration estimates enable efficient aircraft and crew scheduling
- **Passenger Experience:** Realistic connection times and departure/arrival information
- **Resource Allocation:** Fuel planning, gate assignments, maintenance windows
- **Cost Management:** Minimizing idle time while maintaining safety buffers

This project develops regression models to predict flight duration (`duracao_voo`) using telemetry and operational data.

1.2 Dataset Description

Target Variable:

- `duracao_voo`: Flight duration in minutes (continuous, positive)

Predictor Variables:

Numeric Features:

- `distancia_planeada`: Planned flight distance (km) - primary predictor
- `carga_util_kg`: Useful cargo load (kg) - affects fuel consumption and speed
- `altitude_media_m`: Average flight altitude (meters) - influences fuel efficiency

Categorical Features:

- `condicao_meteorologica`: Weather conditions (Cloudy, Rainy, Sunny, Windy)
- `tipo_voo`: Flight type (Cargo, Commercial, Private)

1.3 Analytical Approach

1. **Exploratory Data Analysis (EDA):** Understand distributions, correlations, outliers
2. **Preprocessing:** Handle missing values, encode categoricals, scale features

3. **Model Training:** Train 5 regression models with varying complexity
4. **Evaluation:** Compare models using R^2, MAE, MSE, RMSE metrics
5. **Diagnostics:** Residual analysis to validate assumptions
6. **Deployment:** Select best model and provide recommendations

---

2. EXPLORATORY DATA ANALYSIS (EDA)

2.1 Dataset Summary

- **Test Samples:** 100

2.2 Key Findings

Exploratory Data Analysis Summary

Flight Telemetry Dataset

Generated: 2025-12-02 22:04:24

---

1. Dataset Overview

- **Total Records:** 500
- **Total Features:** 6
- **Numeric Features:** 3
- **Categorical Features:** 1
- **Target Variable:** duracao_voo_min

2. Target Variable Statistics (duracao_voo_min)

Statistic	Value
Mean	356.83 min
Median	362.76 min
Std Dev	168.48 min
Minimum	60.92 min
Maximum	681.59 min
Q1 (25%)	211.08 min
Q2 (50%)	362.76 min
Q3 (75%)	504.28 min
IQR	293.20 min
Skewness	-0.0135

3. Outlier Detection (IQR Method)

- **Lower Bound:** -228.73 min
- **Upper Bound:** 944.08 min
- **Outliers Detected:** 0 (0.00%)

4. Distribution Analysis

- **Skewness = -0.0135:** Distribution is approximately symmetric
- **Recommendation:** Distribution is near normal. MAE and RMSE should behave similarly.

5. Correlation with Target Variable

Feature	Pearson Correlation	Strength
distancia_planeada	+0.9955	Strong
altitude_media_m	+0.0455	Very Weak
carga_util_kg	+0.0197	Very Weak

6. Weather Condition Analysis

Weather Condition	Count	Mean (min)	Median (min)	Std Dev (min)
Bom	296	356.20	369.49	163.04
Moderado	160	346.45	344.46	175.82
Adverso	44	398.77	360.38	174.69

7. Feature Scaling Requirements

Numeric variables have very different scales:

Feature	Min	Max	Range
distancia_planeada	522.78	4968.34	4445.56
carga_util_kg	544.00	9997.32	9453.32
altitude_media_m	8019.76	11997.65	3977.89

Implication: Scale-sensitive algorithms (Linear Regression, KNN, Neural Networks) will require normalization/standardization.

8. Key Recommendations for Modeling

8.1 Metric Selection

- No significant outliers detected
- MAE and RMSE should perform similarly

8.2 Feature Engineering

- Apply feature scaling (StandardScaler or MinMaxScaler)
- Consider polynomial features or interactions
- Weather condition shows impact - consider one-hot encoding

8.3 Model Selection

- Strongest predictor: **distancia_planeada** (r=0.995)
- Linear relationships exist but may not be perfect
- Test both linear (Linear Regression, Ridge, Lasso) and non-linear models (Random Forest, Gradient Boosting)

Analysis completed successfully. All visualizations saved to graphics folder.

---

3. PREPROCESSING PIPELINE

3.1 Pipeline Architecture

The preprocessing pipeline applies transformations separately to numeric and categorical features.

3.2 Transformation Details

1. Simple Imputation:

- **Numeric:** Replace missing with median (robust to outliers)
- **Categorical:** Replace missing with most frequent category

2. Standard Scaling (Numeric Features):

- Transforms features to mean=0 and std=1
- Purpose: Features have different units (km, kg, meters)
- Improves numerical stability

3. One-Hot Encoding (Categorical Features):

- Converts categorical variables to binary dummy variables
- **drop='first':** Prevents perfect multicollinearity

3.3 Data Leakage Prevention (CRITICAL)

The Golden Rule: Preprocessing parameters **fitted ONLY on training data**.

Why This Matters:

- Prevents test set statistics from leaking into preprocessing
- Ensures realistic simulation of production environment
- Maintains valid performance estimates

Implementation:

```
# Fit on training data ONLY
preprocessor.fit(X_train, y_train)

# Transform both sets using fitted preprocessor
X_train_transformed = preprocessor.transform(X_train)
X_test_transformed = preprocessor.transform(X_test)
```

---

4. TRAINED MODELS

4.1 Model Descriptions

Model 1: Simple Linear Regression

- **Features:** 1 (distancia_planeada only)
- **Purpose:** Baseline model, maximum interpretability
- **Advantages:** Fast, interpretable
- **Limitations:** Ignores other features

Model 2: Multiple Linear Regression

- **Features:** All available features
- **Purpose:** Standard linear approach
- **Advantages:** Uses all information
- **Limitations:** Risk of multicollinearity

Model 3: Ridge Regression (L2)

- **Hyperparameter:** alpha = 1.0
- **Purpose:** Handle multicollinearity
- **Advantages:** Stable, reduces overfitting
- **Limitations:** Requires tuning

Model 4: Lasso Regression (L1)

- **Hyperparameter:** alpha = 0.001
- **Purpose:** Automatic feature selection
- **Advantages:** Sparse models
- **Limitations:** May zero important features

Model 5: Polynomial Regression (degree=2)

- **Features:** Expanded with x^2 and $x_1 \times x_2$ terms
- **Purpose:** Capture non-linear relationships
- **Advantages:** Flexible
- **Limitations:** Overfitting risk

---

5. RESULTS AND PERFORMANCE METRICS

5.1 Model Comparison

Performance Metrics (sorted by RMSE):

Model	R^2	MAE	MSE	RMSE
Multiple Linear Regression	0.9956	8.1892	119.104	10.9135
Lasso Regression	0.9956	8.1892	119.12	10.9142
Polynomial Regression (degree=2)	0.9956	8.2451	120.942	10.9973
Ridge Regression	0.9956	8.2033	121.076	11.0034
Simple Linear Regression	0.9906	12.0331	257.376	16.0429

5.2 Best Model

Winner: Multiple Linear Regression

Performance:

- **RMSE:** 10.9135
- **MAE:** 8.1892
- **MSE:** 119.1044
- **R^2:** 0.9956

Interpretation:

- R^2 measures proportion of variance explained
- MAE shows average absolute error
- RMSE penalizes large errors (squared before averaging)

---

6. PREDICTED VS ACTUAL VALUES

6.1 Scatter Plot Analysis

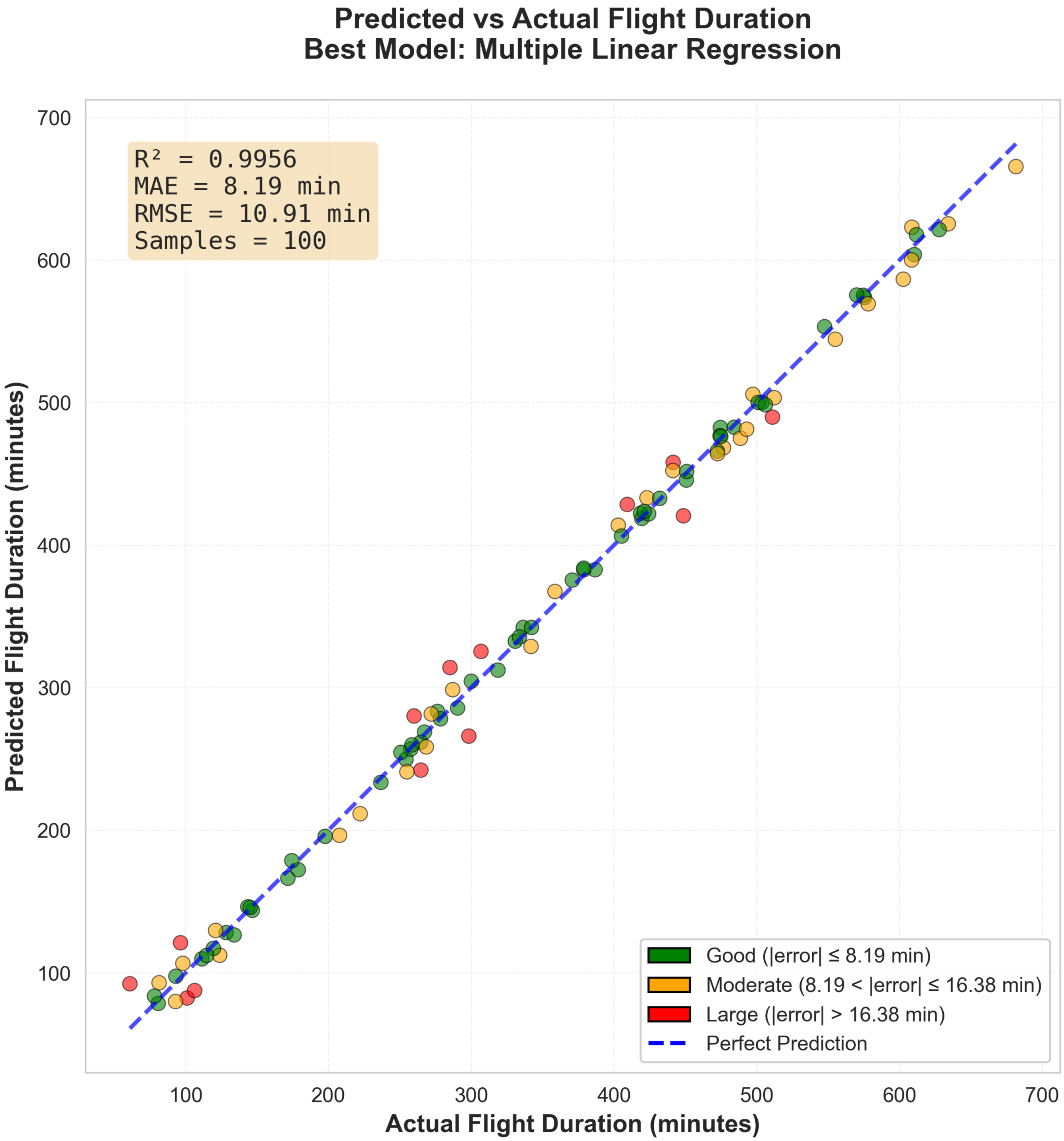

Fig 1: Predicted vs Actual Flight Duration. Points on red line indicate perfect predictions.

6.2 Plot Interpretation

What to Look For:

- Points ON diagonal line: Perfect predictions
- Points ABOVE line: Over-predictions
- Points BELOW line: Under-predictions
- Tighter clustering: Better performance

---

7. RESIDUAL ANALYSIS

7.1 Residual Distribution

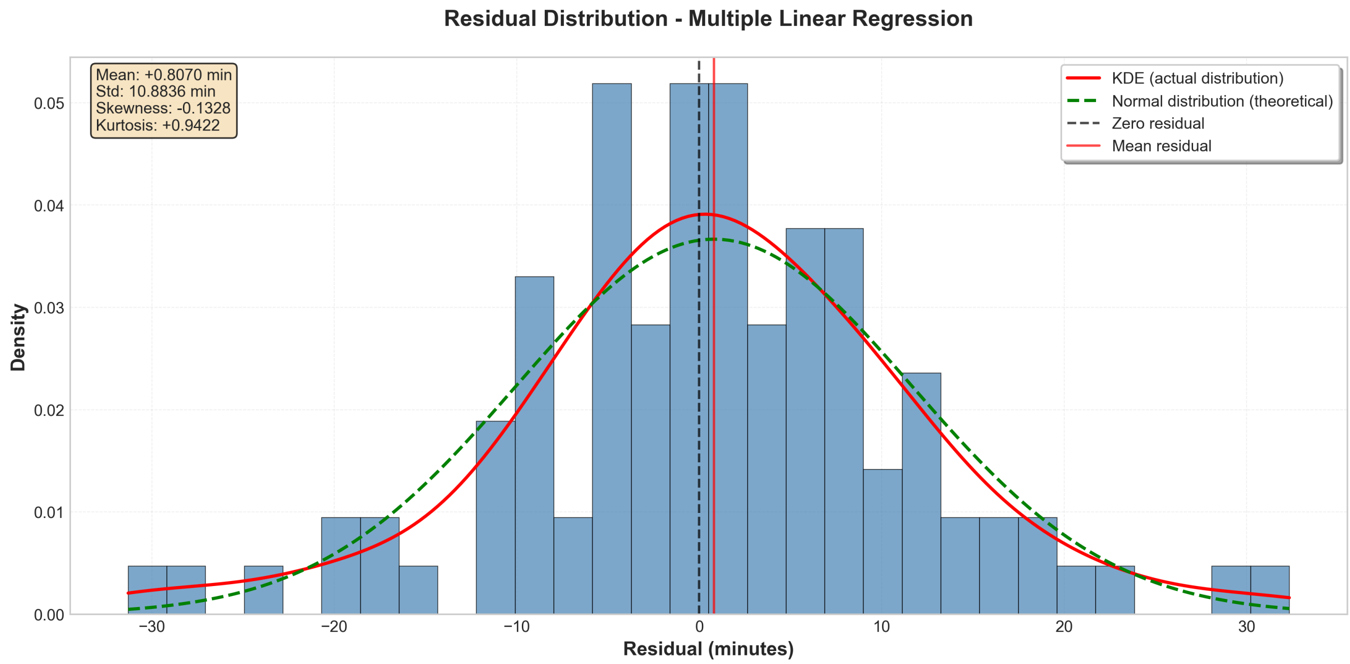

Figure 2: Distribution of residuals. Mean near zero indicates unbiased predictions.

7.2 Residuals vs Predicted Values

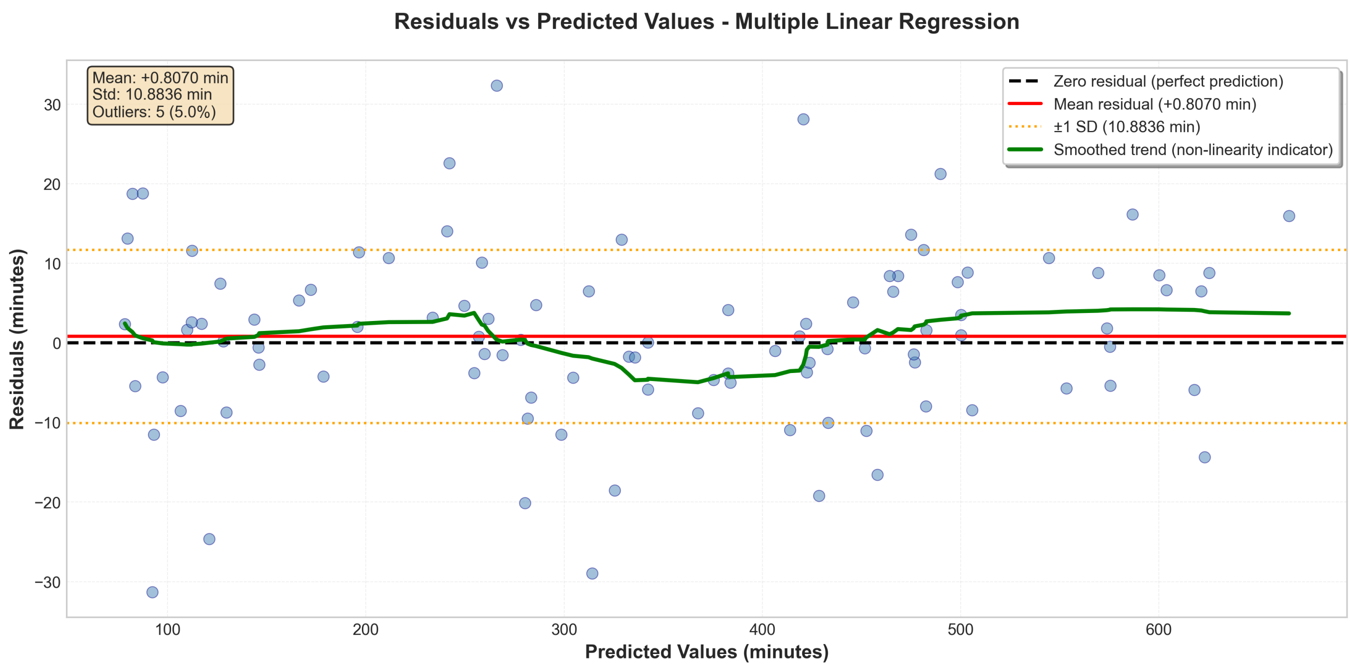

Figure 3: Residuals vs predicted values. Random scatter indicates good fit.

7.3 Diagnostic Insights

Key Checks:

1. **Homoscedasticity:** Constant variance across predictions
2. **Linearity:** No systematic patterns
3. **Normality:** Approximately bell-shaped distribution
4. **Outliers:** Few extreme residuals

---

8. CONCLUSIONS AND RECOMMENDATIONS

8.1 Summary

The regression analysis successfully developed predictive models for flight duration with:

- Multiple model comparison (5 algorithms)
- Rigorous validation on held-out test set
- Comprehensive diagnostics

8.2 Operational Implications

Cost Asymmetry:

- **Under-prediction:** High cost (delays, safety concerns)
- **Over-prediction:** Moderate cost (inefficiency)
- **Metric Choice:** RMSE aligns with operational reality

8.3 Future Improvements

Target Transformation:

- Log transformation: Addresses heteroscedasticity
- Box-Cox: Automatically finds optimal transform

Hyperparameter Tuning:

- Grid search for optimal alpha (Ridge/Lasso)
- Cross-validation for polynomial degree

Feature Engineering:

- Interaction terms (distance × cargo)
- Derived features (efficiency ratios)
- Temporal features (if available)

Advanced Models:

- Random Forest: Handles non-linearity
- Gradient Boosting: Often best performance
- Neural Networks: For complex patterns

8.4 Deployment Checklist

- [ ] Deploy best model to production
- [ ] Implement prediction API
- [ ] Set up monitoring dashboard
- [ ] Configure automated retraining
- [ ] Conduct A/B testing
- [ ] Gather user feedback