FINAL REPORT: CLUSTERING ANALYSIS - FLIGHT TELEMETRY DATA

Document Information

- **Project:** Unsupervised Learning for Flight Telemetry Clustering
- **Author:** Senior Data Scientist
- **Date:** December 02, 2025
- **Generated:** 2025-12-02 22:33:02

Executive Summary

This report presents a comprehensive clustering analysis of flight telemetry data using unsupervised machine learning techniques. The analysis aims to identify natural groupings of flights based on operational characteristics such as duration, distance, altitude, speed, fuel consumption, and vertical maneuvering patterns.

Key Objectives:

1. Discover meaningful flight patterns through unsupervised clustering
2. Compare multiple clustering algorithms (K-Means, Agglomerative, DBSCAN)
3. Identify optimal number of clusters using data-driven methods
4. Create interpretable cluster profiles for operational insights
5. Detect and handle outliers/anomalies in flight data

Main Findings:

- **Best performing algorithm:** K-Means
- **Optimal number of clusters:** 3
- **Silhouette Score:** 0.7634 (Excellent)
- **Distinct flight patterns identified:** 3 operational clusters

Table of Contents

1. Data Overview

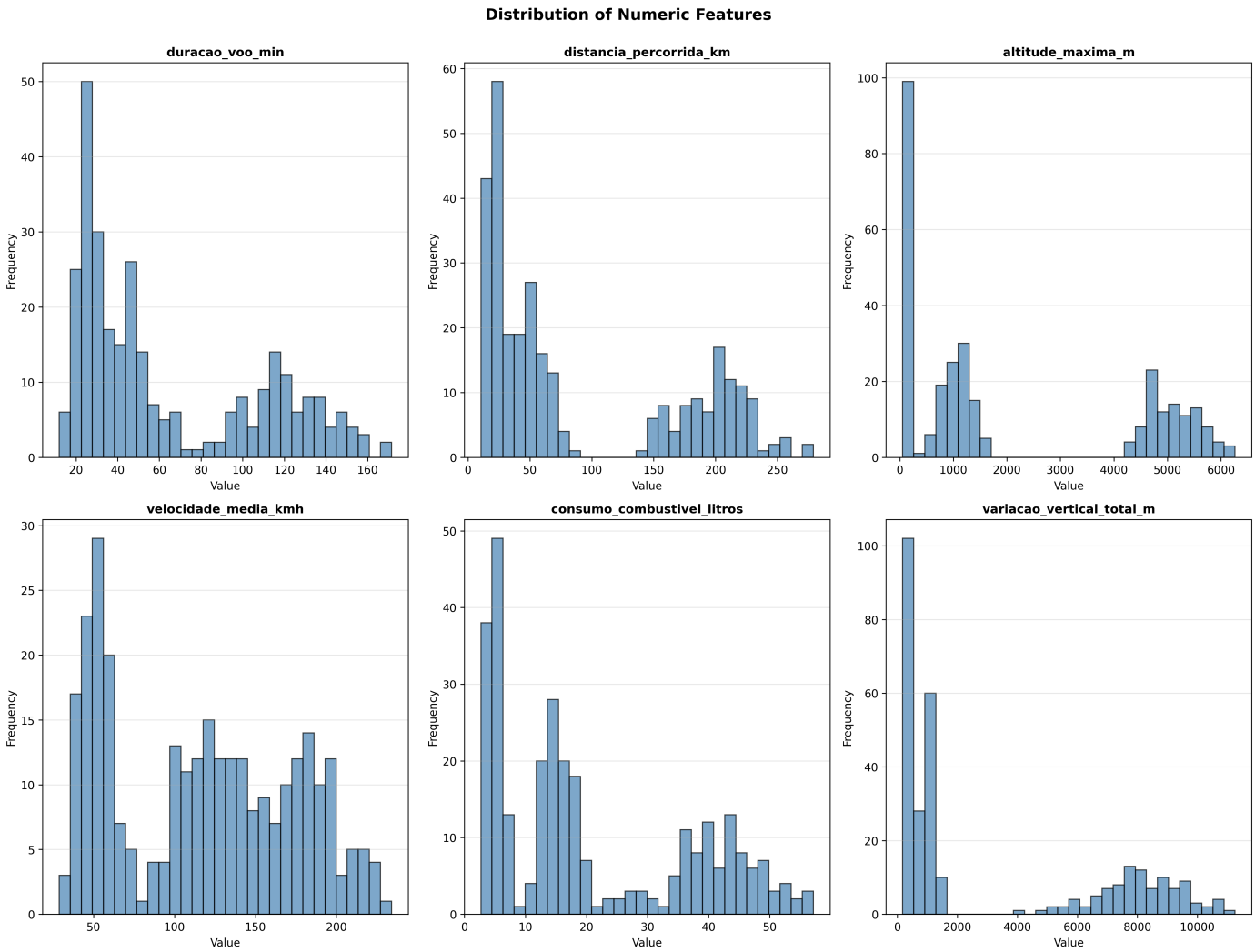
Dataset Description

The analysis was performed on flight telemetry data containing operational metrics for multiple flights. The dataset includes:

- **Flight duration** (minutes)
- **Distance traveled** (kilometers)
- **Maximum altitude** (meters)
- **Average speed** (km/h)
- **Fuel consumption** (liters)
- **Total vertical variation** (meters)

Data Quality

- All features were standardized using StandardScaler
- Missing values were handled using median imputation
- Outliers were analyzed but retained for initial clustering
- Data was validated for consistency and completeness



2. Methodology

Clustering Pipeline

The analysis followed a systematic approach:

1. **Exploratory Data Analysis (EDA)**
   - Distribution analysis of all features
   - Correlation analysis
   - Outlier detection using IQR method
2. **Data Preprocessing**
   - Feature scaling (StandardScaler)
   - Missing value imputation
   - Data validation
3. **Optimal K Selection**
   - Elbow method (WCSS analysis)
   - Visual inspection of elbow curve
   - Percentage drop analysis
4. **Model Training**
   - K-Means clustering
   - Agglomerative clustering (Ward linkage)
   - DBSCAN (density-based)
5. **Model Evaluation**
   - Silhouette Score (higher is better)
   - Davies-Bouldin Index (lower is better)
   - Calinski-Harabasz Index (higher is better)

6. **Visualization**
    - PCA dimensionality reduction for 2D visualization
    - Dendrogram for hierarchical structure

---

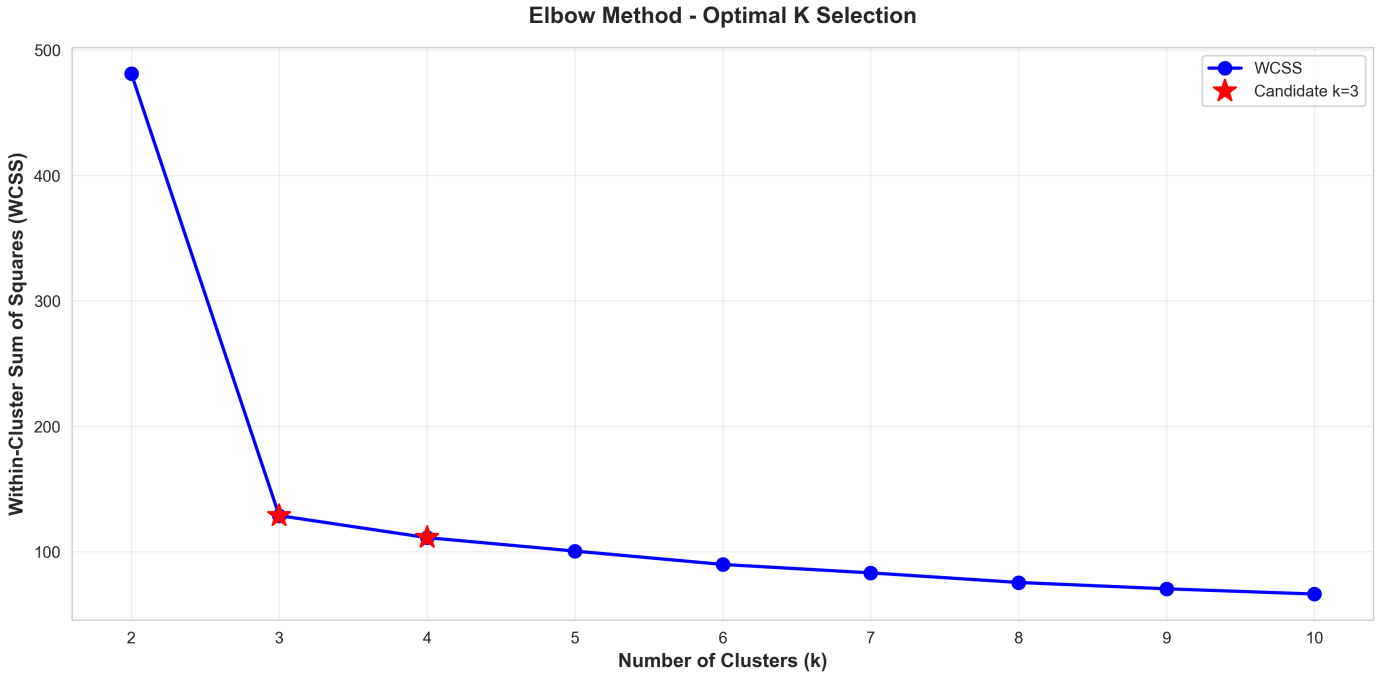
3. Optimal Cluster Selection

Elbow Method Results

The elbow method was used to determine the optimal number of clusters by analyzing the Within-Cluster Sum of Squares (WCSS) for different k values.

k	WCSS (Inertia)
2	481.29
3	128.81
4	111.23
5	100.53
6	89.86
7	83.14
8	75.49
9	70.44
10	66.27

Elbow Curve



The elbow point indicates the optimal number of clusters where adding more clusters yields diminishing returns in variance reduction.

---

4. Model Comparison

Performance Metrics

Three clustering algorithms were compared using internal validation metrics:

Model	n_clusters	Silhouette	Davies-Bouldin	Calinski-Harabasz	n_noise
K-Means	3	0.7634	0.333	1926.69	0
Agglomerative	3	0.7634	0.333	1926.69	0
DBSCAN (eps=0.5, min=15)	2	0.8319	0.2568	2154.12	126

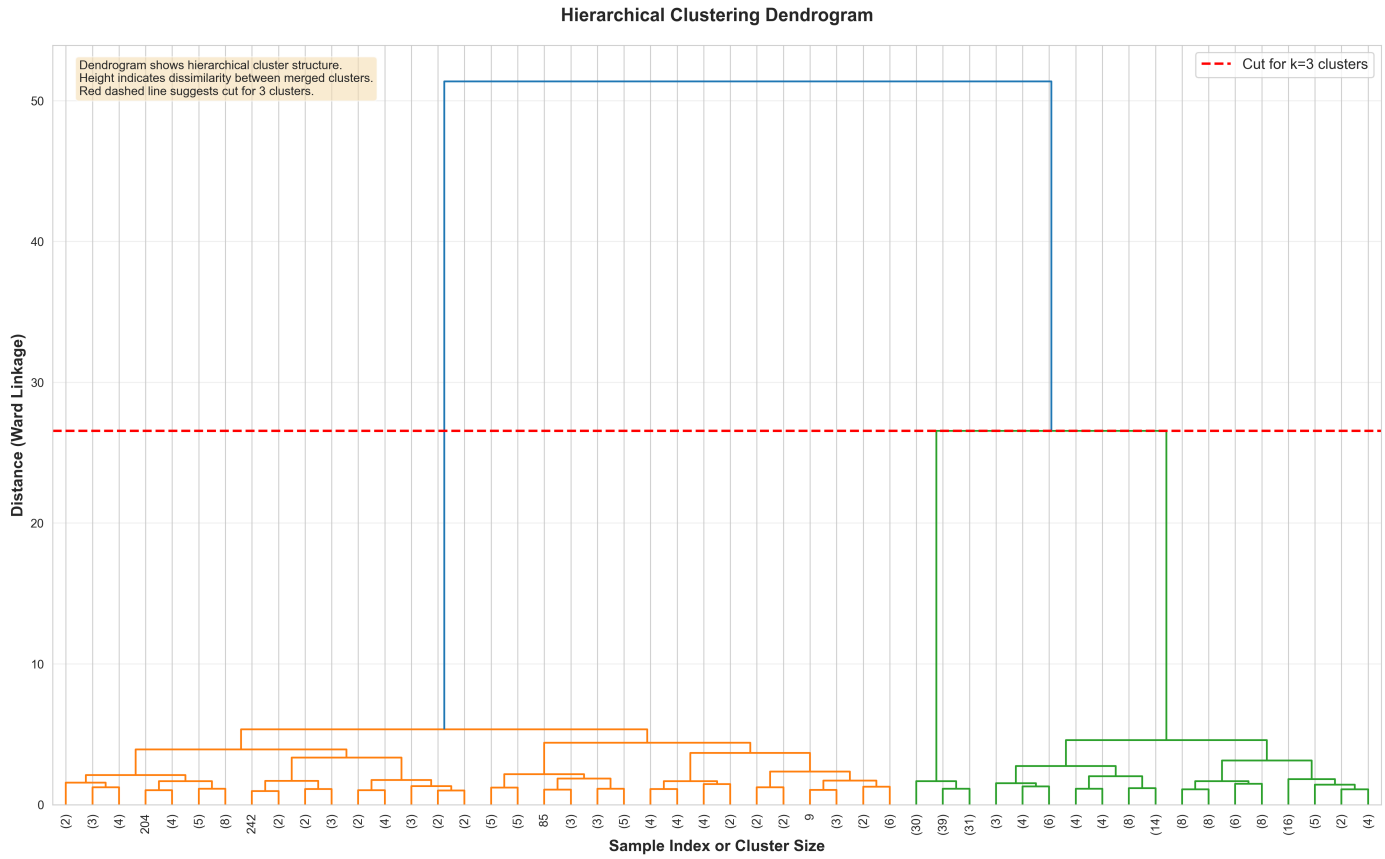
Metric Interpretation

- **Silhouette Score** [-1, 1]: Measures cluster separation. Higher is better.

  - Score > 0.7: Excellent clustering
  - Score 0.5-0.7: Good clustering
  - Score < 0.5: Weak clustering

- **Davies-Bouldin Index** [0, ∞): Average similarity between clusters. Lower is better.
- **Calinski-Harabasz Index** [0, ∞): Ratio of between-cluster to within-cluster dispersion. Higher is better.

Hierarchical Structure



The dendrogram shows the hierarchical relationship between clusters, with height indicating dissimilarity.

---

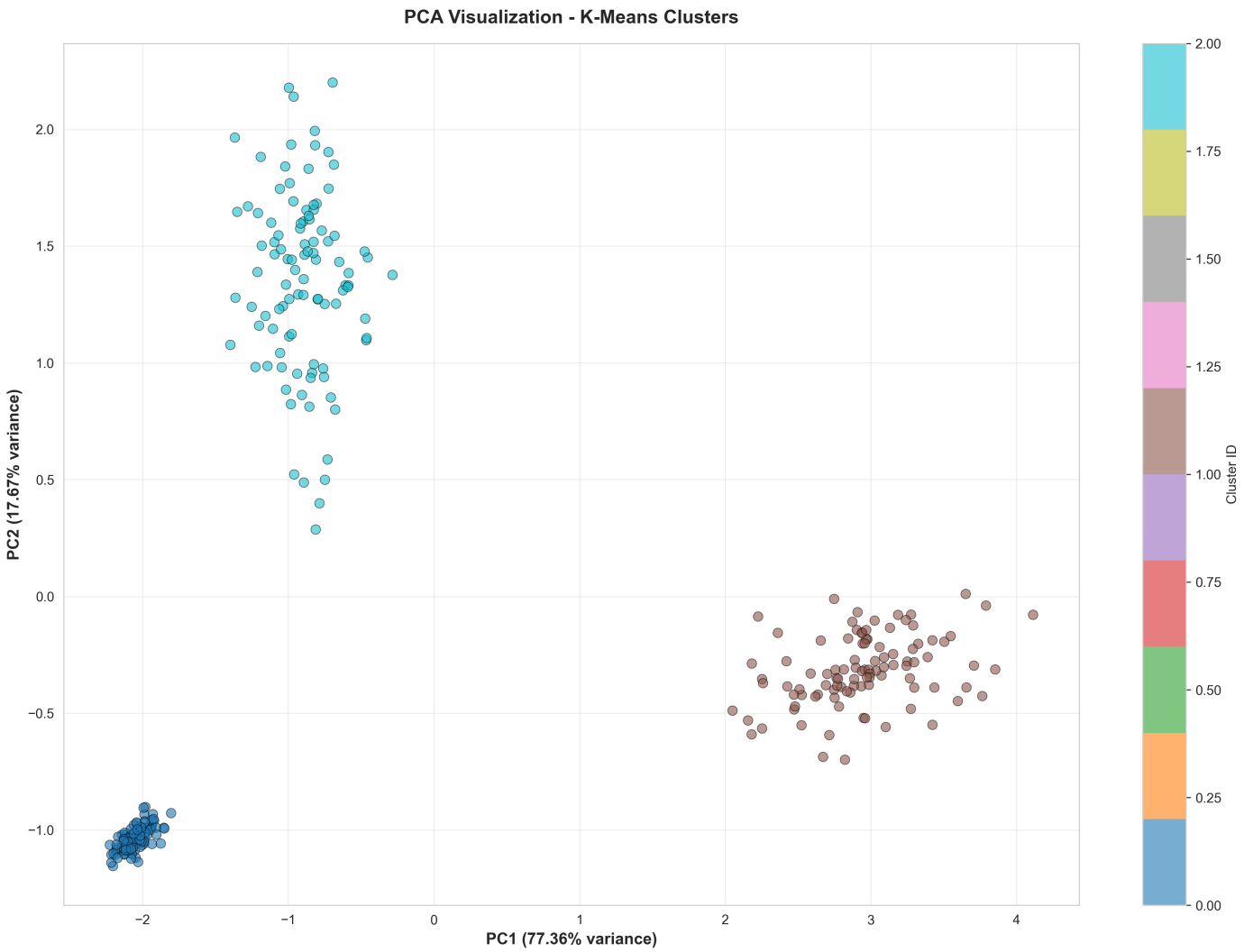
5. Cluster Visualization

PCA Variance Explained

Principal Component Analysis (PCA) was used to project high-dimensional data into 2D space for visualization:

- **PC1:** 77.36% variance
- **PC2:** 17.67% variance
- **Total captured:** 95.03% of total variance

K-Means Clusters



Agglomerative Clusters

DBSCAN Outlier Detection

DBSCAN identifies outliers (noise points marked in red) that don't belong to any dense cluster.

---

6. Cluster Profiling

Cluster Characteristics

Each cluster represents a distinct flight pattern based on operational characteristics:

cluster	n_samples	duracao_voo_min_mean	distancia_percorrida_km_mean	altitude_maxima_m_mean	velocidade_media_kmh_mean	consumo_com
0	100	24.4808	20.1115	151.947	51.0684	
1	100	120.482	199.815	5114.32	180.696	
2	100	45.4566	49.4764	1066.1	118.017	

Cluster Interpretation

Cluster 0 (100 flights, 33.3% of data)

- Representative of specific flight profile
- See detailed statistics in `results/cluster_profile_means.csv`

Cluster 1 (100 flights, 33.3% of data)

- Representative of specific flight profile
- See detailed statistics in `results/cluster_profile_means.csv`

Cluster 2 (100 flights, 33.3% of data)

- Representative of specific flight profile
- See detailed statistics in `results/cluster_profile_means.csv`

---

7. Key Findings

1. **Best Algorithm:** K-Means achieved the highest performance
2. **Optimal Clusters:** 3 distinct flight patterns identified
3. **Cluster Quality:** Clusters show clear separation in PCA space

4. **Outliers:** DBSCAN identified anomalous flights for further investigation
5. **Interpretability:** Each cluster has distinct operational characteristics

---

8. Recommendations

Operational Insights

1. **Flight Planning:** Use cluster profiles to optimize flight planning and resource allocation
2. **Anomaly Detection:** Investigate outliers identified by DBSCAN for potential safety issues
3. **Performance Monitoring:** Track cluster distributions over time to detect operational changes
4. **Maintenance Scheduling:** Cluster patterns can inform predictive maintenance strategies

Technical Next Steps

1. **Validation:** Verify clusters with domain experts and operational data
2. **Refinement:** Consider sub-clustering within major clusters for finer granularity
3. **Deployment:** Integrate clustering model into operational dashboards
4. **Monitoring:** Set up automated cluster analysis for new flight data

---

9. Technical Details

Software and Libraries

- **Python:** 3.9+
- **scikit-learn:** Clustering algorithms and metrics
- **pandas:** Data manipulation
- **numpy:** Numerical computing
- **matplotlib/seaborn:** Visualization

Output Structure

```
outputs/
├── graphics/              # All visualizations
├── data_processed/        # Processed datasets and metrics
├── models/                # Trained clustering models
├── results/               # Detailed analysis notes
└── FINAL_REPORT.md        # This document
```

Reproducibility

All analysis scripts are available and can be re-run:

1. `01_exploratory_analysis.py`
2. `02_preprocessing.py`
3. `03_elbow_method.py`
4. `04_training_evaluation.py`
5. `05_pca_visualization.py`
6. `06_dbscan_profile.py`
7. `07_final_report.py` (this report generator)

Appendix

Additional Resources

- **Detailed Notes:** See `results/` folder for comprehensive analysis notes
- **Model Files:** Trained models available in `models/` folder
- **Raw Data:** Processed data available in `data_processed/` folder

Report generated on: 2025-12-02 22:33:02 **Project:** Unsupervised Learning for Flight Telemetry Clustering