JOURNAL OF AEROSPACE INFORMATION SYSTEMS
Vol. 20, No. 10, October 2023

Study on Machine Learning Methods for General Aviation Flight Phase Identification

Nicoletta Fala*
Oklahoma State University, Stillwater, Oklahoma 74078
Georgios Georgalis†
Tufts University, Medford, Massachusetts 02155
and
Nastaran Arzamani‡
Oklahoma State University, Stillwater, Oklahoma 74078

https://doi.org/10.2514/1.I011246

Accurate identification of phases of flight is an essential step in analyses such as airport operation counts, fuel burn estimation, and safety studies. Past research has focused primarily on using positional data with rule-based or probabilistic-based decision-making to identify the phases of flight. Many of these efforts note that the task of correctly identifying phases of flight is challenging, often requiring extreme fine-tuning of methods. In this paper, we initially study whether combinations of dimensionality reduction of flight data records from general aviation aircraft impact clustering into the correct flight phases (climb, cruise, or descent) without any preprocessing or fine-tuning. For dimensionality reduction, we considered the low variance filter, the high correlation filter, principal component analysis, and autoencoders. We found that these dimensionality reduction algorithms do not offer any benefit for the phase identification task, as compared to feature selection that simply omits engine-specific features. For the clustering task, we considered *K*-means and Gaussian mixture models. After performing clustering on eight test flights, we conclude that both methods are adequate at identifying the phases of flight in various general aviation flights and yield similar results.

I. Introduction

IDENTIFICATION of phases of flight is an essential step in flight data monitoring. Flight phase information is necessary for airport operation counts [1], fuel burn modeling and estimation [2,3], and safety analyses [4]. Identifying phases of flight is the first step in aircraft performance modeling, air traffic analyses and simulation, airspace capacity studies, as well as continuous descent approach analyses. In safety studies, adverse events need to be accompanied by the phase of flight where the event occurred to allow the analyst to determine the severity of the situation. Additionally, this flight phase identification process can help us in detecting aircraft faults. For example, the phase of flight is an input to a fuel burn analysis that varies based on operational phase: the aircraft burns more fuel during the climb portion as compared to the descent (because it operates at a higher-power setting). A flight attitude may be considered unsafe during one phase of flight but safe in another, such as a high bank angle during the cruise/maneuvering portion as opposed to the same attitude during the final approach segment. Although having knowledge of the correct flight phase is important, obtaining this information during postflight processing is not trivial because of the challenge of defining where one phase ends and another starts, especially in general aviation (GA). Technology in the flight deck has provided us with a variety of flight data from GA operations, which are captured via onboard flight data recorders, Automatic Dependent Surveillance-Broadcast (ADS-B), or other personal devices. However, studies on identifying phases of flights have been mainly conducted on commercial operations. Whereas commercial flights tend to have very prescribed and uniform operations, GA operations are freeform in nature. That is, they may climb to an altitude, cruise for some time, maneuver, and change altitude irregularly before climbing to a different altitude. Additionally, the use of autopilot is not as prevalent in GA, and the range of altitude in a flight is smaller, resulting in small deviations in altitude that make it difficult to define where one phase begins and another ends.

One of the requirements in identifying phases of flight is that the flight segments are mutually exclusive and exhaustive. In other words, they completely identify the contents of the entire flight, with each data point only being labeled for one phase of flight. By defining accurate constraints on each phase of flight, we are able to assign each segment of the flight to a specific unique flight phase. Table 1 defines each phase of flight based on publications of the International Civil Aviation Organization (ICAO) [5] and the National Transportation Safety Board (NTSB) [6], and it provides some constraints used in past research for each fight phase.

Past research has focused on primarily two approaches to identify the phases of flight from flight data records: logical rule-based decision-making, and probabilistic-based decision-making. In fuel burn studies, the cruise segment was defined as any period of time where the engine is operating at less than the maximum revolutions-per-minute (RPMs) setting, the indicated airspeed is higher than the stall speed, and the groundspeed is positive, with the aircraft operating at an altitude of at least 3000 ft above ground level (AGL) [7]. This definition is only useful for fuel burn models, however, because using it for general identification of the flight phase would result in any aircraft cruising below the 3000 ft AGL threshold not being classified as being in the cruise phase of flight. Goblet et al. proposed altitude-based rules in their models to identify the phases of flight for Cirrus SR20 aircraft, but they noted that such approaches can have a large variance in the percentage of misidentifications, depending on the specific flight characteristics [8]. Zhang et al. developed a fuzzy logic and Toeplitz inverse covariance-based clustering unsupervised machine learning methodology that uses ADS-B data to identify phases of flight to be further used in improved noise and emissions models [1]. As they mentioned, relying on ADS-B data may be

Received 30 January 2023; revision received 12 May 2023; accepted for publication 23 June 2023; published online Open Access 11 August 2023.

*Assistant Professor, Mechanical and Aerospace Engineering, 201 General Academic Building; nfala@okstate.edu. Member AIAA (Corresponding Author).

†Data Scientist, Data Intensive Studies Center, 177 College Ave.; georgios.georgalis@tufts.edu. Member AIAA.

‡Graduate Research Assistant, Mechanical and Aerospace Engineering, 201 General Academic Building. Student Member AIAA.

Table 1 Definitions of phases of flight found in the literature

Phase of flight	ICAO [5]	NTSB [6]	Goblet et al. [8]
Standing	Before pushback or taxi; or after arrival at the gate, ramp, or parking area while the aircraft is stationary	Any time before taxi or after arrival while the aircraft is stationary	Ground speed $= 0$ kt of indicated air speed (KIAS) and indicated airspeed $= 0$ KIAS
Taxi	The aircraft is moving on the aerodrome surface under its own power before takeoff or after landing	The aircraft is moving on the ground before takeoff and after landing	Ground speed <25 KIAS and indicated airspeed $= 0$ KIAS
Takeoff	From the application of takeoff power, through rotation, and to an altitude of 35 ft above runway elevation	From the application of takeoff power, through rotation and to an altitude of 35 ft above runway elevation	Engine RPMs >2500 and altitude <35 ft above the runway elevation
Climb	From the end of the takeoff subphase to the first prescribed power reduction, or until reaching 1000 ft above runway elevation or the visual flight rules (VFRs) pattern: whichever comes first	Any time the aircraft has a positive rate of climb for an extended period of time	Change in altitude (constant increase >200 ft/ min)
Cruise	Cruise: Any level-flight segment after arrival at initial cruise altitude until the start of descent to the destination	The time period following the initial climb during which the aircraft is in level flight	Altitude constant ($\pm$200 ft/ min)
Descent	Instrument flight rules (IFRs): Descent from cruise to either initial approach fix or VFRs pattern entry. VFRs: Descent from cruise to the VFRs pattern entry or 1000 ft above the runway elevation: whichever comes first	Any time before approach during which the aircraft has a negative rate of climb for an extended period of time	Change in altitude (constant decrease < -200 ft/ min)
Approach	IFRs: From the initial approach fix to the beginning of the landing flare. VFRs: From the point of VFRs pattern entry, or 1000 ft above the runway elevation, to the beginning of the landing flare	From the point of pattern entry, or 1000 ft above the runway elevation, to the beginning of the landing flare	Altitude constant decrease (< -200 ft/ min) and altitude <1000 ft above runway elevation
Landing	From the beginning of the landing flare until aircraft exits the landing runway, comes to a stop on the runway, or when power is applied for takeoff in the case of a touch-and-go landing	From the beginning of the landing flare until the aircraft touches down and exits the landing runway, or comes to a stop on the runway, or when power is applied for takeoff, depending upon the intended action after landing	Indicated airspeed <60 KIAS and altitude constant decrease < -200 ft/ min for 1 min

limited due to discontinuous, inaccurate, or missing signals. Kuzmenko et al. [9] and Zhang and Mott [10] also used positional data and probabilistic approaches to classify aircraft in different flight phases. However, such methods may be limited because they require assumptions that may not always be correct for GA operations (e.g., that there is equal probability of being in a specific flight phase) apart from relying on potentially inaccurate ADS-B data.

Chati and Balakrishnan used latitude, longitude, pressure altitude, ground speed, and their slopes to identify the transition points between flight phases in an Airbus A330; and they categorized flight phases into departure taxi, takeoff roll and wheels off, ascent/climb, cruise, descent, touchdown, and arrival landing roll and taxi [11]. Gong and McNally introduced a methodology for the automated statistical analysis of trajectory prediction accuracy as a function of the level-flight, climb, or descent phases of flight using 2774 large commercial jet flights [12]. Paglione and Oaks used tracked radar surveillance data to determine the horizontal phase of flight (i.e., whether the aircraft was flying straight or turning) and the vertical phase of flight (i.e., whether the aircraft was flying level, ascending, or descending) [13]. To build pilot advisory systems, Kelly and Painter applied fuzzy sets to identify flight phases. Flight segment identification (FSI) identified real-time flight phases and operational procedures of an aircraft by monitoring aircraft state parameters and flight events [14]. They considered taxi, takeoff, climbout, cruise, initial approach, final approach, and landing. Their FSI (which was an enabler to pilot advising, procedure guidance, and automated highways in the sky) was based on fuzzy sets and is categorized into the state-based FSI (which uses the flight parameters to determine the flight operations currently executed by the pilots) and the procedural FSI (which identifies the flight segments the pilot should be flying) [14]. As part of a larger project aiming to build aircraft performance models, Sun et al. used ADS-B data to identify phases of flight using air traffic management data by applying clustering with the balanced iterative reducing and clustering using hierarchies method and the density-based spatial clustering of applications with noise method as well as fuzzy logic: each of which were employed to solve different levels of the identification problem [15]. They categorized flight phases into ground, climb, descend, cruise, and unidentifiable states by using the altitude, rate of climb, and ground speed as inputs [15]. Because the data points are relatively close to their neighbors based on the Euclidean distance between time stamps, altitudes, velocities, and positions (and due to differences between aircraft types and their divergent flight procedures), clustering alone without fuzzy logic may not be as successful [16]. Kovarik et al. applied three artificial intelligence models [long short-term memory (LSTM) recurrent neural network, neural ordinary differential equations, and support vector machine) to develop three different phases of flight forecasting machine learning models using a four-dimensional Global Positioning System (GPS) and radar track data from 57 flights provided by an en route computer system [17]. Tian et. al used aircraft simulator data to categorize different phases of flight [18]. Their work points out that classifying flight processes using flight parameters is useful in performing the state quality evaluation of the whole flight but also helps in detecting aircraft faults. They used a decision tree classifier to split the flight in into takeoff, climb, cruise, descent, and landing phases by using the altitude, pitch angle, airspeed, and aircraft axial acceleration as input variables [18]. Arts et al. also used simulated flight data and employed a combination of K-means clustering with an LSTM network for identifying phases of flight [19]. Chin et al. applied various supervised machine learning techniques to identify phases of flight for rotorcraft operations to pave the way for anomaly detection analyses, and they presented a demonstrative dashboard with statistics about phases of flight, which may be insightful in designing other platforms [20]. Liu et al. used Flight Data Recorder (FDR) data by employing a flight phase recognition method based on a Gaussian mixture model (GMM) for detecting abnormal events as well as improving system safety and reliability. In this work, they used real FDR data for commercial airliners from NASA Open Data Portal,[§] to analyze the effectiveness of their approach [21].

From these efforts, it is evident that the task of correctly identifying phases of flight is challenging, often requiring extreme fine-tuning of methods from flight to flight. The goal of this paper is to study whether and how various combinations of machine learning algorithms perform for the task of flight identification from data records without detailed fine-tuning or human intervention in labeling flights.

[§]https://data.nasa.gov/.

The methods we use here are combinations of two sequential processes: reducing dimensionality, and clustering. Reducing dimensionality aims to transform the data from the original high-dimensional space to a lower-dimensional space, where we assume the low-dimensional representation allows for the important properties of the data to be maintained, the weak features to be filtered, and the data be less sparse. Clustering describes the unsupervised learning task of assigning data points (the flight data observations) to groups or clusters (the flight phases).

In this work, we use four prominent methods for dimensionality reduction: the low variance filter (LVF), the high correlation filter (HCF), principal component analysis (PCA), and autoencoders (AEs). The LVF is a feature selection algorithm where the decision to keep or discard a feature is based only on the within-feature variance after normalization. Features with low variance are removed because including them is assumed to not affect the target task, given the feature spread is very small. The HCF is a sequential selection algorithm where features are added, given the constraint that a feature cannot be added if it is highly correlated with an already selected feature, as measured by the Pearson correlation coefficient. The decision threshold is an input to the HCF process. PCA is an orthogonal linear transformation of the data into a lower dimension. The first component is the basis vector of the coordinate system in the lower dimension, along which the variance is maximized when the data are projected onto that component. Projecting on the second component yields the second maximum variance, and so on. AEs are neural networks that have two parts: the encoder that maps the input to a lower-dimension space, and the decoder that reconstructs the input from the representation space. The middle layer between the encoder and the decoder resembles a *bottleneck*; i.e., it forces the feature embedding to be in a lower-dimensional space than the original feature space. AEs are designed to be able to reconstruct the dataset from the lower representation, but they can also be used for dimensionality reduction as we do here by using the lower-dimensional representation after the bottleneck.

For clustering, we used the K-means algorithm and Gaussian mixture models. K-means is a distance-based algorithm that clusters data by minimizing the within-cluster distances between the observations and the respective cluster centroids. The GMM clusters data by assuming that data in the same cluster belong to the same Gaussian distribution and that the dataset is a Gaussian mixture of these distributions.

The first part of the paper tests the hypothesis of whether dimensionality reduction in general is necessary for flight phase identification. To answer, we use the four aforementioned dimensionality reduction techniques on a set of test flights and evaluate the clustering results with three quantitative metrics: the silhouette score (or SC), the Calinski–Harabasz index (or CHI), and the Davies–Bouldin index (or DBI). All metrics measure the similarity within and between clusters to evaluate the clustering result. The second part of the paper uses the results of the previous process to qualitatively demonstrate and evaluate the clustering result while including a new approach to improve the flight phase identification via postprocessing (PP) using segment detection.

The paper is organized as follows: Section II describes the process to collect the flight data records. Section III includes a discussion on the methods we used for dimensionality reduction of the flight data, namely, the low variance filter, the high correlation filter, principal component analysis, and autoencoders. Section IV describes the methods for clustering the flight data representations, namely, K-means and Gaussian mixture models. In Sec. V, we test whether dimensionality reduction is necessary for flight phase identification. Section VI includes a deeper analysis using qualitative evaluation and improvement of the flight identification results for eight test cross-country flights of two GA aircraft. Section VII provides recommendations for future research. Section VIII summarizes the paper.

II. Flight Data Records Collection

General aviation aircraft do not have the same recording capabilities we have come to expect of commercial airliners fitted with "black boxes." However, newer aircraft tend to be equipped with glass-cockpit avionics, i.e., screens with computers in the background that use the atmospheric and system inputs to reconstruct the information the pilot would traditionally see on "steam gauge" avionics. Some of these avionics suites provide the pilot or operator with a copy of the data displayed.

The datasets used in this paper come from the Garmin integrated flight instrument displays on the airplanes in Oklahoma State University's (OSU's) fleet. The OSU Flight Center uses Cessna 172 airplanes (Skyhawks) for primary training and Cirrus SR20 G6 airplanes for advanced training. The Cessnas available come in two models: C-172R and C-172S. All airframes are equipped with similar avionics and can provide similar data: the Cessna 172 with Garmin G1000, and the SR20 with Cirrus Perspective Plus systems. Both variants have FDR capabilities; they record all data fields displayed on the screen on an SD card easily accessible to the user. The G1000 logs flight data in a comma separated values (.csv) file, with the top rows dedicated to airframe, Garmin hardware and software information, and headers; and it generates a new .csv file with each engine start. The records contain information gathered from the aircraft's GPS unit, air data computer, attitude and heading reference system, communication and navigation radios, and engine monitoring system in a tabular format at a frequency of ∼1 Hz. The FDR records 64 and 80 parameters for the C-172 and SR-20, respectively.

A. Data Collection

For this research, we have used datasets from C-172 and SR20 aircraft. These datasets were recorded from all kinds of training flights: some involve laps around the pattern for landing practice, others are flights to the practice areas for training and practice on flight maneuvers, some are flights for instrument approaches, and (lastly) some are cross-country flights. In this research, we are focusing on the cross-country flights, which are broadly defined as flights where the aircraft took off from one airport and navigated to a different destination. These flights may include a touch and go followed by the return flight to the airport of origin, or they may include a full-stop landing and engine shutdown at the destination: in which case, the return flight is on a different dataset. Additionally, some of the datasets may be corrupt or unusable, for example, if the pilot shut down the engine shortly after starting it without flying. These flights are easily identifiable through their path and altitude plots. We use the graphical user interface shown in Fig. 1 to visualize the flights and quickly classify them as a cross-country flight (XC), a maneuvers flight (MNV), a flight which includes instrument approaches (APP), or a flight of "unknown" classification (UKN), i.e., a flight that cannot easily or accurately be classified using the two provided plots. Using this classification, we take a random sample of 79 cross-country flights: 63 from the C-172, and 16 from the SR20 aircraft.

B. Flight Data Variables

The flight phase identification autoencoder hyperparameters will likely differ for the various aircraft models due to design

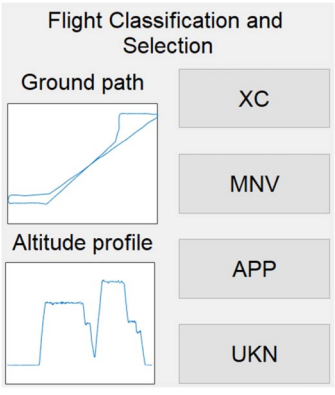

Fig. 1 The graphical interactive user interface to select flight records that will be included in the training and testing datasets.

Table 2 Random sample of available flight data observations from a Cessna 172S cross-country flight

Altitude, ft	IAS, kt	VS, ft/min	Pitch, deg	RPMs	CHT1, F	CHT2, F	CHT3, F	CHT4, F	EGT1, F	EGT2, F	EGT3, F	EGT4, F
1076.5	85.85	737.06	8.4	2481.8	329.53	321.27	331.53	325.11	1320.24	1319.87	1311.68	1290.95
1668.6	91.65	692.24	6.9	2511.7	368.08	362.97	361.22	365.36	1323.92	1328.24	1320.2	1299.63
2108.5	94.12	894.66	6.9	2527.6	381.58	376.67	371.65	382.01	1320.35	1325.92	1315.76	1294.17
3329.4	94.59	528.3	5.75	2533.7	397.78	390.08	384.21	404.85	1293.35	1296.29	1292.57	1272.78
3484.1	107.22	−6.28	0.88	2462	376.01	364.7	367.18	393.71	1462.17	1456.93	1445.97	1431.11

differences. The Cirrus SR20 G6 has a 215 hp engine; the Cessna 172 engine is rated for 160 or 180 hp for the R and S models, respectively. The SR20 is, overall, a much faster airplane than the C-172. As a result, the SR20 will climb at a higher vertical speed and airspeed. The climb and descent profiles will therefore not look the same for each airplane, creating an additional challenge when it comes to using machine learning to cluster the flight phases. Table 2 shows sample observations for the parameters of interest to this project. We expect that the parameters listed will help identify the operational phase of flight at each data point. Prior work relied on altitude to identify phases of flight [8]. Indicated airspeed, however, will also change when climbing or descending, depending on the power settings used. Airspeed will decrease when climbing and increase when descending, assuming the power is not changed drastically. Pilots, however, will make use of their power to intentionally climb or descend. Pitch will increase when climbing and decrease when descending. Lastly, the air-cooled engine will not get as much air through the inlet when climbing, resulting in increased temperatures.

The flight datasets are preprocessed as follows: First, any missing observations (due to instrumentation lag) and any data corresponding to a nonmoving aircraft [indicated air speed (IAS) = 0] are removed. The temperatures that are reported from all four cylinders (both for the cylinder heat and the exhaust) are not independent; therefore, we only include the mean value per cylinder for the cylinder head temperature (CHT) and the exhaust gas temperature (EGT). Additionally, apart from using the altitude directly, we also include the difference of the altitude to the previous time step, with the initial value being zero. We expect that the altitude as a value itself does not carry sufficient information about the flight phase, whereas the altitude change does. In addition to the raw value for the vertical speed, we also include a moving average for the vertical speed over 20 time steps; the initial data points use a moving average of the previous time steps that are available. We expect that the vertical speed at a time step is useful, but a moving average carries more reliable information about the flight phase (e.g., if an aircraft is on a continuous climb, then the moving average of the vertical speed will be considerably larger than if the aircraft is on a non-autopilot cruise). Lastly, we normalize the data using the min-max normalization to remedy effects from the different feature units. In total, we consider nine features: altitude, IAS, vertical speed (VS), pitch, RPMs, CHT_m, EGT_m, difference in altitude from the previous time step (dAltitude), and VS_{MA}.

III. Dimensionality Reduction of Flight Data

There are four main and major issues with flight data records that make phase identification challenging: there exist nonlinear relationships between the different available features (Fig. 2), they are on different scales with varying degrees of correlation between them (Fig. 3a), and they are time dependent and discontinuous (Fig. 4b). Given these challenges with the dataset, attempting a clustering effort directly on the dataset with an off-the-shelf algorithm (e.g., centroid based or density based) would perhaps not yield good results. In this paper, we test whether preprocessing the data using dimensionality reduction algorithms improves the clustering result. We use the LVF, the HCF, PCA, and a deep autoencoder to generate representations of the flight data features; and then we use the representations to identify the phases of flight rather than the flight data as is. The dimensionality reduction is considered separately for the test flights because they can be of different airframes, engines, airports, and flight patterns. For the LCF, we remove any of the nine features that have a smaller variance than the average variance of all the features; so, feature i will be removed if $\sigma_i^2 < \overline{\sigma_i^2}$. For the HCF, we start from the original set of nine features and sequentially remove features that have a Pearson's coefficient larger than 0.6. For PCA, we use the algorithm in the *sklearn* package for Python, to maintain 90% of the variance in the representation. For the autoencoder, we created a deep neural network trained to reconstruct the flight data $\boldsymbol{X}$ in the original feature space $\mathcal{S}^D$ after projecting the data into a nonlinear manifold repre-

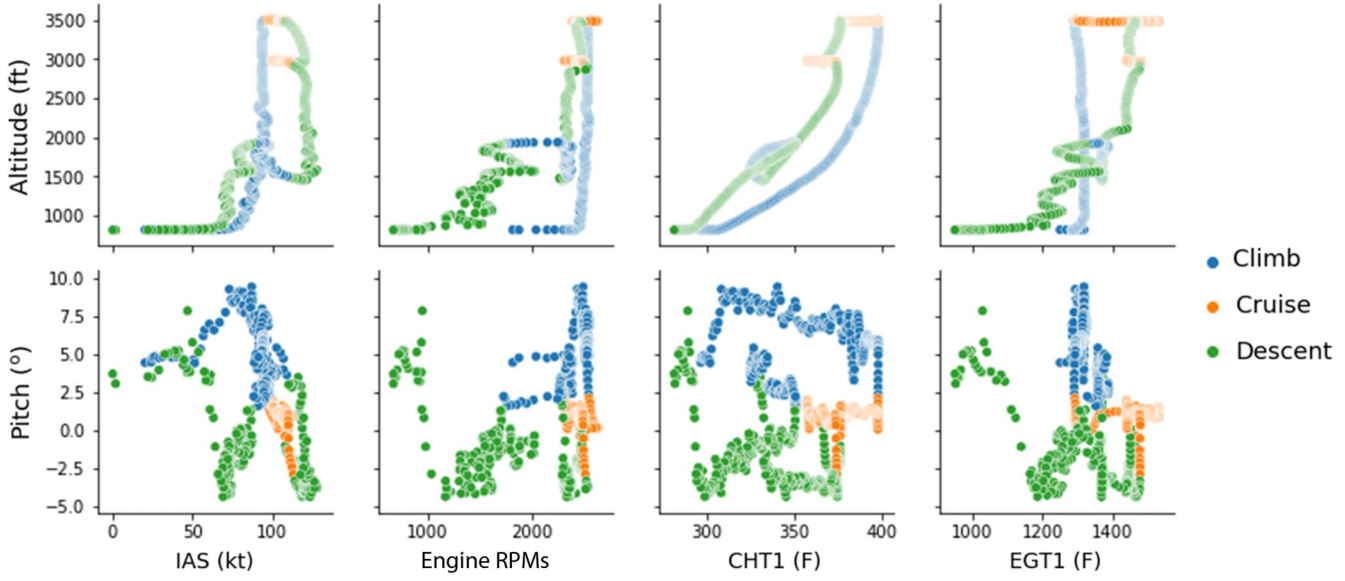

Fig. 2 Flight data feature relationships from an example Cessna 172S cross-country flight.

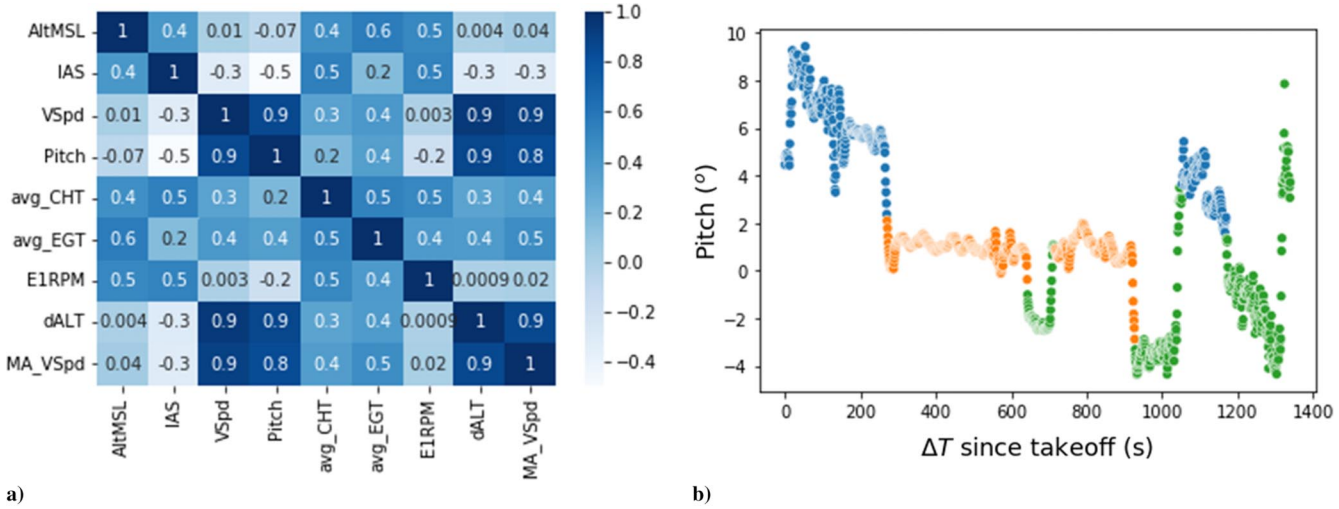

Fig. 3 Correlation map between available flight data features (Mean Sea Level altitude or AltMSL, indicated airspeed or IAS, vertical speed or VSpd, pitch, average cylinder head temperature or avg_CHT, average exhaust gouge temperature or avg_EGT, engine rotations per minute or E1RPM, difference in altitude from the previous time step or dALT, and a moving average of the vertical speed or MA_VSpd) from all available cross-country flights in training set (Fig. 4a). Pitch angle time series from an example Cessna 172S cross-country flight showing abrupt changes and discontinuity (Fig. 4b).

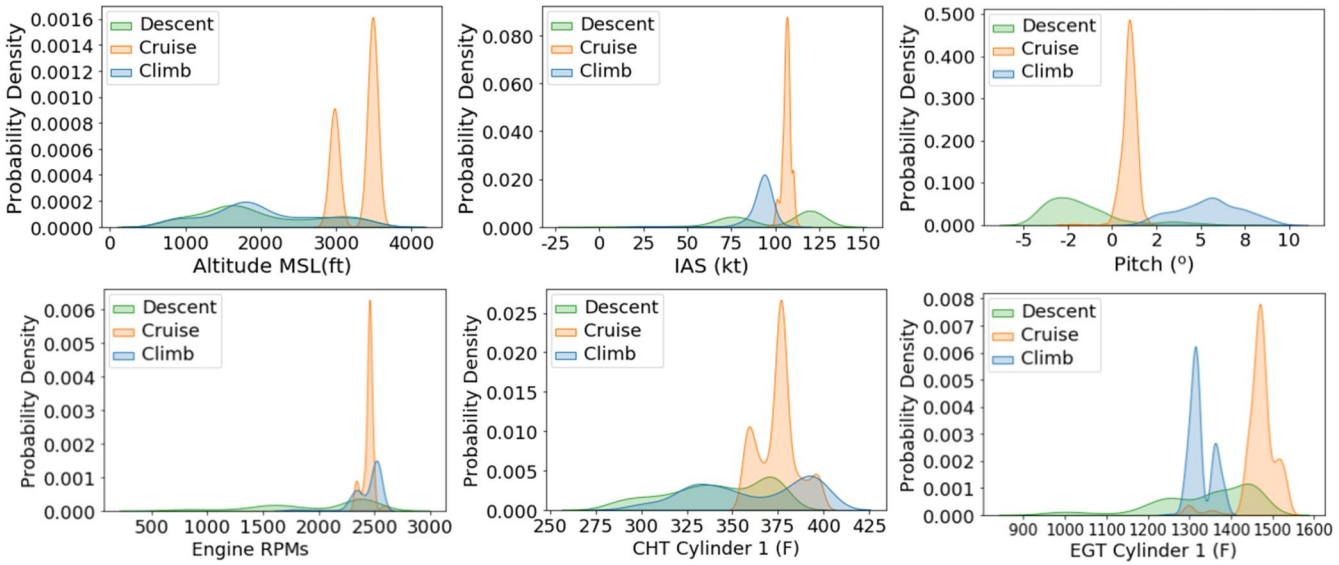

Fig. 4 Probability distributions of flight data records from an example Cessna 172S cross-country flight. MSL denotes mean sea level.

sented via the embedding vector $\boldsymbol{v} \in \mathcal{S}^d$ [22]. For our available flight data, the feature dimension is $D = 9$ and the latent representation dimension is $d = 4$. After training the autoencoder (i.e., find the weights and bias of all its layers), we use the low-dimensional latent variables $\boldsymbol{Z}$ for the flight phase identification that follows.

The autoencoder has the following architecture. The encoder includes five fully connected layers: each with batch normalization and a scaled rectified linear unit [scaled exponential linear unit (SeLU)] activation function. Each of the θ_i sequential layers has $n_{\theta_i} = n_{\theta_{i-1}} - 1$ neurons (i.e., one less neuron than the previous layer until the bottleneck). The decoder is symmetrical to the encoder, and consequently includes five fully connected layers with batch normalization, SeLU activation functions, and an equal number of neurons for the corresponding layers. Using the SeLU as the activation function performs better at normalizing the network as compared to other activation functions for this dataset (i.e., each layer preserves the mean and variance from the previous layer) [23]. The architecture is shown in Fig. 5 together with the overall workflow of methods for the paper. We built the architecture in *tensorflow*, and the network is trained via backpropagation with the Adam optimizer with an adaptive learning rate. The objective is to minimize a mean squared error (MSE) loss function:

$$\mathcal{L} = \frac{1}{N}\sum_{i=1}^{N}(x_i - \hat{x}_i)^2$$

where $N = 363{,}768$ is the total number of flight data observations in the training set, x_i is the ith observation vector in the training set, and $\hat{x}_i$ is the corresponding reconstructed vector from the autoencoder. The training stops after the loss is not improved for 30 consecutive epochs with a tolerance of e^{-5}. Four randomly selected flights from each aircraft were kept as the testing/validation set. A graph of the loss progression during training is shown in Fig. 6.

IV. Clustering Using K-Means and Gaussian Mixture Models

We use clustering to group the latent representations of flight data generated by the autoencoder architecture or K-means, with the

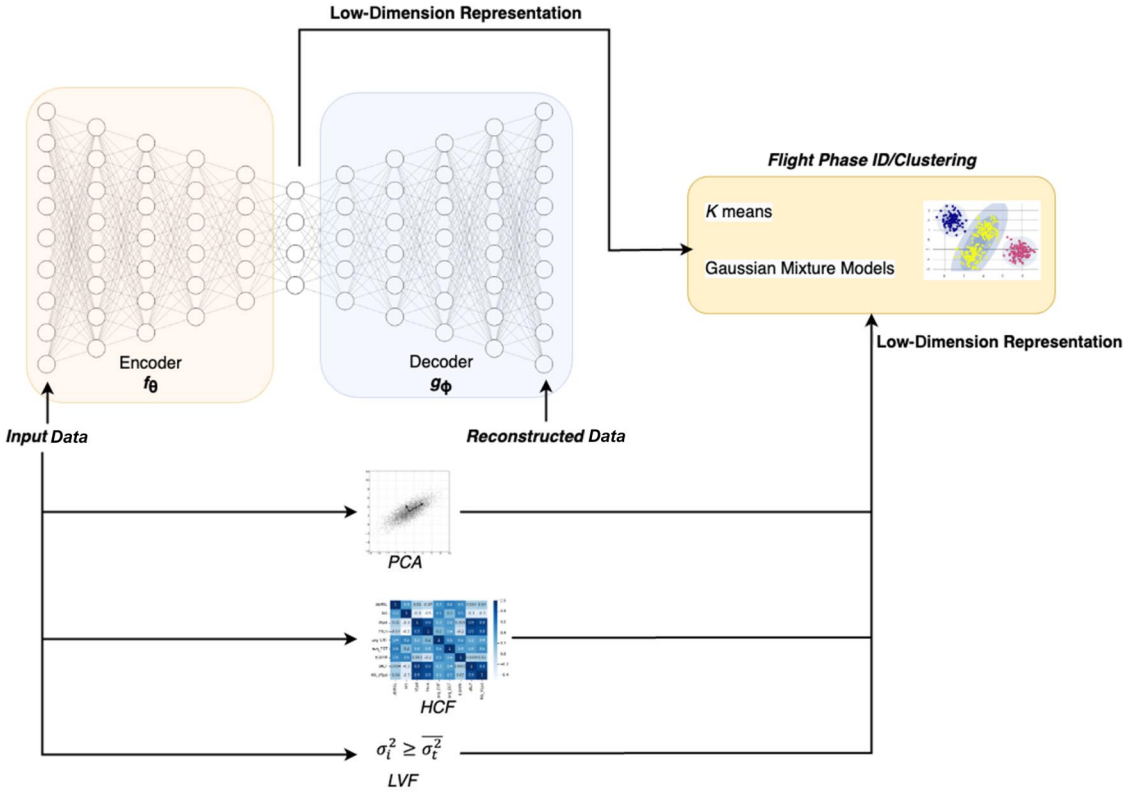

Fig. 5 The machine learning methods used in this work. The low-dimension representations are clustered for flight phase identification (Flight Phase ID).

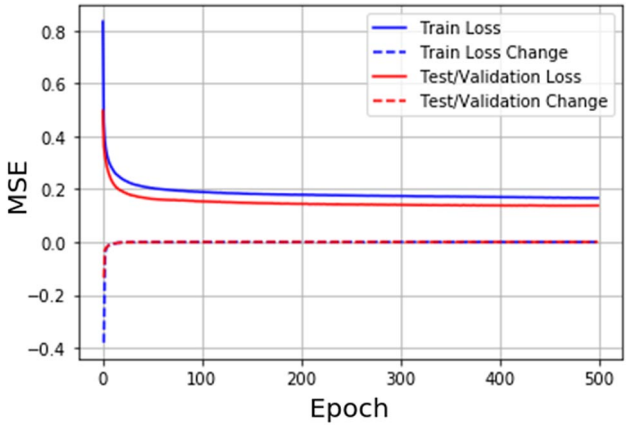

Fig. 6 MSE loss during the training of the autoencoder used for this work.

assumption that these groups or clusters correspond to the correct flight phase: climb, cruise, or descent. Clustering is an unsupervised machine learning task, which is the appropriate approach because we do not have phase identification labels for the flight data. Clustering algorithms rely on similarity metrics to decide which data points should belong to the same group or cluster. There exist different algorithms for clustering in the literature (see Ref. [24] for an exhaustive list). In general, clustering methods have significant variability in the quality of their results and are data dependent [25]. In this paper, we assume that Gaussian mixture models are more appropriate for identifying flight phases for two reasons:

1) They allow soft clustering (each data observation vector x_i belongs to flight phase cluster C_i with some probability p_i).

2) They assume that the flight data are generated from a mixture of Gaussian distributions, which holds for the flight data records during each flight phase [6], based on the modes of the distributions [26]. Soft clustering may be more appropriate for this task, given that for some data points, it may be able to communicate the level of ambiguity that the data point belongs to a particular flight phase or another. For comparison, we also show the clustering results with the K-means algorithm [27].

Under the assumption that the flight data come from a Gaussian mixture with K distinct Gaussian distributions and corresponding means μ_k, covariance matrices Σ_k and proportion weights π_k, the marginal distribution of an observation is then

Table 3 Quantitative metrics for K-means clustering of low-dimension representation of flight data from low variance filter and high correlation filter

	SC			CHI			DBI		
Flight no./aircraft	LVF	HCF	Benchmark	LVF	HCF	Benchmark	LVF	HCF	Benchmark
1 (C-172)	0.422	0.388	0.494[a]	826.12	664.78	1,000.03[a]	0.889	0.943	0.683[a]
2 (C-172)	0.567	0.543	0.660[a]	4,037.11	3,867.28	7,098.53[a]	0.734	0.705	0.599[a]
3 (SR-20)	0.778	0.730	0.790[a]	6,659.13	4,860.24	10,825.2[a]	0.683	0.638	0.623[a]
4 (SR-20)	0.412	0.365	0.527[a]	2,139.28	2,036.62	3,578.15[a]	0.951	0.953	0.874[a]
5 (C-172)	0.538	0.527	0.490	2,818.98	3,245.66	3,780.89[a]	0.962	0.939	0.878[a]
6 (C-172)	0.375	0.372	0.441[a]	2,113.54	2,057.73	3,396.65[a]	1.050	1.039	0.927[a]
7 (SR-20)	0.581	0.429	0.644[a]	8,538.54	2,615.85	11,050.3[a]	0.603	1.013	0.504[a]
8 (SR-20)	0.646	0.584	0.706[a]	7,053.8	3,087.13	8,699.7[a]	0.616	0.942	0.568[a]

[a]Best clustering result for given flight, metric, and dimensionality reduction method.

Table 4 Quantitative metrics for *K*-means clustering of low-dimension representation of flight data from principal component analysis and autoencoders

	SC			CHI			DBI		
Flight no./aircraft	PCA	AE	Benchmark	PCA	AE	Benchmark	PCA	AE	Benchmark
1 (C-172)	0.455	0.420	0.494[a]	847.80	723.385	1,000.03[a]	0.707	0.774	0.683[a]
2 (C-172)	0.582	0.540	0.660[a]	3,815.06	4,368.63	7,098.53[a]	0.628	0.752	0.599[a]
3 (SR-20)	0.795[a]	0.795[a]	0.790	7,760.26	11,977.8[a]	10,825.2	0.665	0.626	0.623[a]
4 (SR-20)	0.411	0.417	0.527[a]	2,218.72	2,367.67	3,578.15[a]	0.976	0.901	0.874[a]
5 (C-172)	0.567[a]	0.460	0.490	3,398.61	3,007.18	3,780.89[a]	0.866[a]	0.951	0.878
6 (C-172)	0.428	0.406	0.441[a]	2,658.83	2,414.75	3,396.65[a]	0.920[a]	0.945	0.927
7 (SR-20)	0.531	0.609	0.644[a]	3,950.44	9,509.92	11,050.3[a]	0.836	0.592	0.504[a]
8 (SR-20)	0.665	0.625	0.706[a]	5,747.62	6,220.57	8,699.7[a]	0.680	0.707	0.568[a]

[a]Best clustering result for given flight, metric, and dimensionality reduction method,

Table 5 Quantitative metrics for GMM clustering of low-dimension representation of flight data from low variance filter and high correlation filter

	SC			CHI			DBI		
Flight no./aircraft	LVF	HCF	Benchmark	LVF	HCF	Benchmark	LVF	HCF	Benchmark
1 (C-172)	0.422[a]	0.370	0.339	817.41[a]	613.524	596.02	0.903[a]	0.964	1.397
2 (C-172)	0.474	0.523	0.555[a]	2585.20	3184.47[a]	3037.97	0.926	0.759[a]	1.339
3 (SR-20)	0.731	0.749[a]	0.679	4867.03	3576.18	6266.95[a]	0.865[a]	0.913	1.006
4 (SR-20)	0.308	0.212	0.362[a]	1313.59[a]	1056.75	1048.1	1.206[a]	1.500	1.596
5 (C-172)	0.359	0.425[a]	0.380	1351.53	2191.41	2546.32[a]	1.841	1.321	1.230[a]
6 (C-172)	0.344	0.320	0.391[a]	1902.81	1602.65	2656.56[a]	1.110[a]	1.126	1.157
7 (SR-20)	0.408	0.472	0.500[a]	2176.11	2455.19	2786.87[a]	1.360	1.063[a]	1.212
8 (SR-20)	0.587	0.568	0.649[a]	4692.93	2742.39	7088.11[a]	0.792	0.990	0.661[a]

[a]Best clustering result for given flight, metric, and dimensionality reduction method.

Table 6 Quantitative metrics for GMM clustering of low-dimension representation of flight data from principal component analysis and autoencoders

	SC			CHI			DBI		
Flight no./aircraft	PCA	AE	Benchmark	PCA	AE	Benchmark	PCA	AE	Benchmark
1 (C-172)	0.359[a]	0.340	0.339	660.967[a]	603.06	596.02	1.076	0.991[a]	1.397
2 (C-172)	0.433	0.455	0.555[a]	2247.94	2092.73	3037.97[a]	1.037[a]	1.642	1.339
3 (SR-20)	0.709[a]	0.675	0.679	4977.12	5182.8	6266.95[a]	0.900[a]	1.032	1.006
4 (SR-20)	0.358	0.360	0.362[a]	1666.83	1792.29[a]	1048.1	1.089	1.022[a]	1.596
5 (C-172)	0.320	0.320	0.380[a]	1343.57	1999.14	2546.32[a]	2.011	1.460	1.230[a]
6 (C-172)	0.389	0.232	0.391[a]	2205.74	1280.06	2656.56[a]	1.003[a]	1.339	1.157
7 (SR-20)	0.379	0.565[a]	0.500	2526.86	8087.43[a]	2786.87	0.925	0.650[a]	1.212
8 (SR-20)	0.564	0.527	0.649[a]	2878.2	3457.37	7088.11[a]	0.946	0.995	0.661[a]

[a]Best clustering result for given flight, metric, and dimensionality reduction method.

Table 7 Qualitative evaluation criteria for the clustering of the flight phases

Evaluation criteria	Does not satisfy ($\times = 0.0$)	Partially satisfies ($\sim = 0.5$)	Satisfies ($\surd = 1.0$)
A. Does the method correctly cluster climb?	Climb phase(s) is not separated from cruise/descent.	Climb phase data points exist incorrectly within other flight phases, or parts of climb phases are not separated from cruise/descent.	All data points that should be classified as climb are correctly classified as such, and no data points from a different phase are classified as climb.
B. Does the method correctly cluster cruise?	Cruise phase(s) is not separated from climb/descent.	Cruise phase data points exist incorrectly within other flight phases, or parts of cruise phases are not separated from climb/descent.	All data points that should be classified as cruise are correctly classified as such, and no data points from a different phase are classified as cruise.
C. Does the method correctly cluster descent?	Descent phase(s) is not separated from climb/cruise.	Descent phase data points exist incorrectly within other flight phases, or parts of descent phases are not separated from climb/cruise.	All data points that should be classified as descent are correctly classified as such, and no data points from a different phase are classified as descent.
D. Does the method correctly identify transient points in the data (i.e., identifies the moment of change in phase of flight rather than continuing with the prior phase into the next one)?	No transient phases detected despite other methods picking them up.	Some transient phases detected correctly.	All transient phases detected correctly.
E. Does the method correctly identify step climb/descent (i.e., identifies short level-offs during a longer descent phase)?	Level-offs are not separated from climb/descent.	Some level-offs are misidentified as climb/descent.	All level-offs are correctly separated from climb/descent.

$$P(\boldsymbol{X_i} = \boldsymbol{x}) = \sum_{k=1}^{K} P(C_i = k) P(\boldsymbol{X_i} = \boldsymbol{x} | C_i = k)$$
$$= \sum_{k=1}^{K} \pi_k N\left(\boldsymbol{X_i}; \mu_k, \sigma_k^2\right) \quad (1)$$

where $C_i \in \{1, 2, \ldots, K\}$ is the latent variable representing the mixture component for X_i, $P(\boldsymbol{X_i} = \boldsymbol{x} | C_i = k)$ is the mixture component, and $P(C_i = k) = \pi_k$ is the mixture proportion weight (which is the probability that X_i belongs to mixture component C_i). $N(\mu_k, \sigma_k^2)$ represents the normal distribution of mixture component K with mean μ_k and variance σ_k^2.

The parameters of the Gaussian mixture are computed using the expectation maximization algorithm [28], which starts with some initialization for the Gaussian parameters

$$\omega_0 = \left\{\mu_{k_0}, \Sigma_{k_0}, \pi_{k_0}\right\}$$

to find the posterior distribution of the latent variables $P(C|X, \omega_0)$ and update to $\hat{\omega}$ by maximizing the log-likelihood $L_{\log}(\omega, \omega_0)$:

$$\hat{\omega} = \text{argmax}_{x_\omega}\ L_{\log}(\omega, \omega_0) \quad (2)$$

$$L_{\log}(\omega, \omega_0) = E_{C|X_i, \omega_0}[\log P(\boldsymbol{X_i}, C|\omega)]$$
$$= \sum_{k=1}^{K} P(C = k|\boldsymbol{X_i}, \omega_0) \log(P(\boldsymbol{X_i}, C|\omega) \quad (3)$$

To evaluate the clustering results for the different dimensionality reduction methods, we use three quantitative metrics: the silhouette score, the Calinski–Harabasz index, and the Davies–Bouldin index. The silhouette score is defined as the mean silhouette coefficient $s(i)$ for each sample i in the data:

$$s(i) = \begin{cases} 1 - a(i)/b(i) & \text{if } a(i) < b(i) \\ 0 & \text{if } a(i) = b(i) \\ b(i)/a(i) - 1 & \text{if } a(i) > b(i) \end{cases} \quad (4)$$

where $a(i)$ is the mean Euclidean distance between i and any other data point in the same cluster, and $b(i)$ is the smallest mean Euclidean

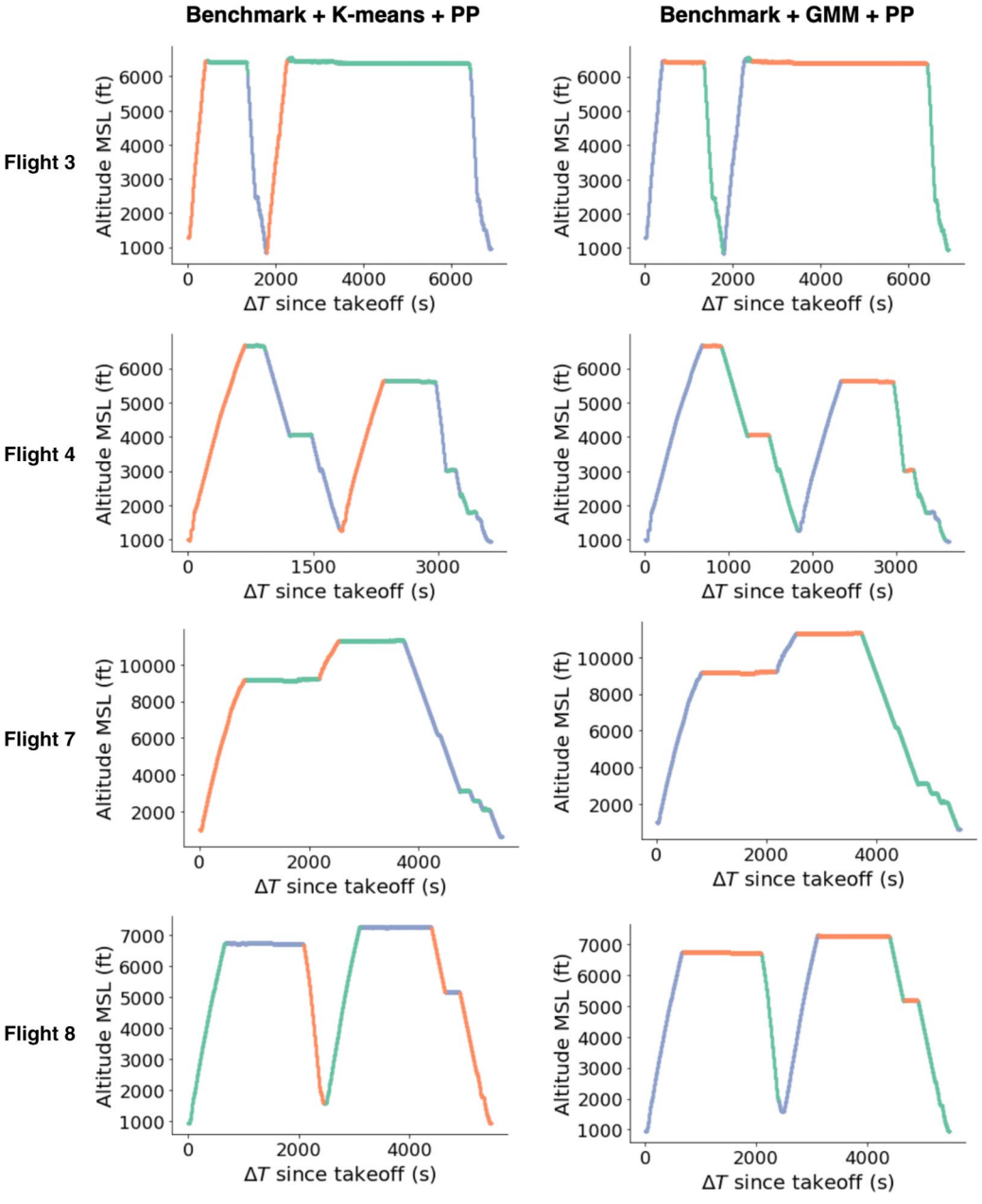

Fig. 7 Clustering results for the four test flights of SR-20 aircraft.

distance between i and any other data point in a different cluster. Given these definitions, it follows that $-1 \leq SC \leq 1$, with a larger SC corresponding to a better clustering result: the average dissimilarity between points in the same cluster is minimized, whereas the neighboring cluster dissimilarity is maximized.

The Calinski–Harabasz index is the ratio of the sum of intercluster dispersion and the sum of intracluster dispersion for all clusters as follows:

$$\text{CHI} = \frac{B_k}{W_k} \times \frac{N-K}{K-1} \tag{5}$$

$$B_k = \sum_{k=1}^{K} n_k \times \|C_k - C\|^2 \tag{6}$$

$$W_k = \sum_{k=1}^{K} \sum_{i=1}^{n_k} \|X_{ik} - C_k\|^2 \tag{7}$$

where $K = 3$ is the number of clusters, N is the total number of observations in the flight, n_k is the number of observations within cluster k, C_k is the centroid of cluster k, C is the centroid of the entire flight dataset, and X_{ik} is the ith observation in cluster k. Comparatively, a higher CHI corresponds to a better clustering result.

The Davies–Bouldin index is defined as the average similarity measure of each cluster with its most similar cluster, i.e., the ratio of within-cluster distances to between-cluster distances:

$$\text{DBI} = \frac{1}{K} \sum_{i=1}^{K} \max_{j \neq i} \frac{S_i + S_j}{M_{i,j}} \tag{8}$$

where S_i is a similarity measure for cluster i, and $M_{i,j}$ is a measure of separation between two neighboring clusters i and j:

$$S_i = \frac{1}{n_k} \sum_{i=1}^{n_k} \|X_{ik} - C_k\|^2 \tag{9}$$

$$M_{i,j} = \|C_i - C_j\| \tag{10}$$

where n_k the number of observation within cluster k, C_* is the centroid of a cluster, C is the centroid of the entire flight dataset, and X_{ik} is the ith observation in cluster k. Comparatively, a lower DBI corresponds to a better clustering result.

V. Evaluation of Dimensionality Reduction Methods

In this section, we first test whether and how the different combinations of dimensionality reduction shown in Fig. 4 impact the clustering results using quantitative metrics. We tested these methods on a total of eight flights: four from the SR-20 aircraft, and four from the C-172 aircraft. The data records from these flights were not used in training the AE model. Tables 3–6 include the quantitative evaluation of the clustering result for each combination of flight, the clustering method, and the dimensionality reduction method. The tables also include a benchmark that is the best clustering result we could achieve by making feature selection based on subject-matter expertise instead of algorithms by removing engine-specific features because the aircraft have different engines and power for a total of six features. Based on these results, we observe that for most combinations of flights/metrics/clustering techniques, the benchmark feature selection outperforms the dimensionality reduction from algorithms such as the LVF, the HCF, PCA, or AEs. There are some exceptions for some specific cases and combinations, but the benefits seem marginal overall. We therefore conclude that for the flight data we studied in this paper, algorithms such as the LVF, the HCF, PCA, or AEs are unnecessary and do not offer any benefit; and a set of features that is engine independent is better overall.

VI. Phase of Flight Identification Results on Test Flights

Our previous analysis suggests that we should use six features for our clustering (dAltitude, IAS, VS, pitch, VS_{MA}, and altitude). Therefore, we use these six features with the same clustering techniques (K-means and GMM) and study the quality of the flight identification in detail. We have also added a postprocessing approach using segment detection to improve these results further. Because we have no "truth" labels for the phases of flights, we rely on qualitative criteria to evaluate the accuracy of the flight phase identification for the two methods. These criteria aim to provide an assessment on whether the method successfully detected a particular segment, whether it identified potentially transient points in the flight, and whether the correct clustering could be easily obtained via simple postprocessing, as shown in Table 7.

The postprocessing algorithm (Algorithm 1) works as follows: Given a clustering result, it first searches for the length of the clustered segments and their locations in the data. After the segments have been identified, a threshold is set so that if a segment has fewer

Algorithm 1: Postprocessing algorithm for flight identification using segment detection

Given a flight data point X_i, the corresponding cluster labels $X_{i,c}$, and j_s as the length of segment s, the first task is identifying the location of the segments s:

For $i = [0{:}size(X_i)]$, **do**:
  If $X_{i,c} = X_{i+1,c}{:}\, j \leftarrow j + 1$
  If $X_{i,c} \neq X_{i+1,c}{:}\, s \leftarrow s + 1, j \leftarrow 0, \text{keep}(j, s)$
End for

After having identified the segment locations s, a threshold is set for small segments to have their labels replaced with larger neighboring segments:

For $i = [0{:}size(s)]$, **do**:
  If $j_s \leq threshold$:
    $k_1 = \{s | s > i \;\&\; j_s > 30\}$
    $k_2 = \{s | s < i \;\&\; j_s > 30\}$
    If $X_{k_1,c} = X_{k_2,c}{:}X_{i:j,c} \leftarrow X_{k_1,c}$
    If $X_{k_1,c} \neq X_{k_2,c}{:}X_{i:j,c} \leftarrow max_j(X_{k_1,c}, X_{k_2,c})$
End for

Table 8 Total scoring based on qualitative criteria for all SR-20 test flights

	Criterion	Benchmark+ K-means + PP	Benchmark+ GMM + PP
Flight 3	A: Climb	√	~
	B: Cruise	√	√
	C: Descent	√	~
	D: Transient	~	~
	E: Steps	√	√
Flight 4	A: Climb	√	~
	B: Cruise	~	√
	C: Descent	√	√
	D: Transient	~	√
	E: Steps	√	√
Flight 7	A: Climb	√	√
	B: Cruise	√	√
	C: Descent	√	√
	D: Transient	√	√
	E: Steps	√	√
Flight 8	A: Climb	√	√
	B: Cruise	√	√
	C: Descent	√	√
	D: Transient	√	√
	E: Steps	√	√
Total scoring out of 20	— —	18.5	18

consecutive points than the threshold, the labels of that segment are flipped according to larger neighboring segments that are above the set threshold. The threshold should be tuned, and it should be small enough to ensure large segments are not mislabeled. We set the threshold to 30 data points; this is to only relabel small segments in the data that may have been misclassified.

The flight phase identification results for the SR-20 aircraft are shown in Fig. 7. Flight 3 includes two sequences of continuous climb–cruise–descent at the same altitude end points. Both approaches work well for this flight. GMM detected the transient point after the second climb. *K*-means continued classifying some of the descents following the cruise as cruise, whereas the aircraft was clearly descending. Flight 4 is more challenging to cluster than the previous two; it includes two cruise phases at different altitudes with long step-resembling descents. Both the *K*-means and the GMM cluster these phases well in general. *K*-means mislabels part of the descent in the last segment together with the small segment that follows, where the aircraft is holding altitude. The GMM is also not perfect because it groups the last segment where the aircraft is holding the altitude with the initial part of the descent that follows. Overall, flight 4 is well clustered. Flight 7 includes two main cruise phases with a long continuous stepped descent at the end. The *K*-means correctly labels the entire flight, whereas the GMM does not explicitly separate the very brief segments of constant altitude during the descent. Both of these approaches would be acceptable by a flight instructor, given that the small segments of constant altitude are in reality transient points rather than actual cruise phases. Flight 8 includes three distinct cruises phases with smooth climbs and descents. Both methods provide an excellent and equal result. The scoring for the SR-20 flights is shown in Table 8.

The flight phase identification results for the C-172 aircraft are shown in Fig. 8. The C-172 flights do not have autopilot, which is evident by the constant variations in altitude. Flight 1 is very challenging to correctly segment, given that it includes a very long climb at a varying rate. Both the *K*-means and the GMM cluster the entire flight correctly, grouping the initial points as the climb with the following descents. The flight includes a small segment that could be considered a low-altitude cruise, which is also identified correctly. Flight 2 includes two cruise phases at similar altitudes with some

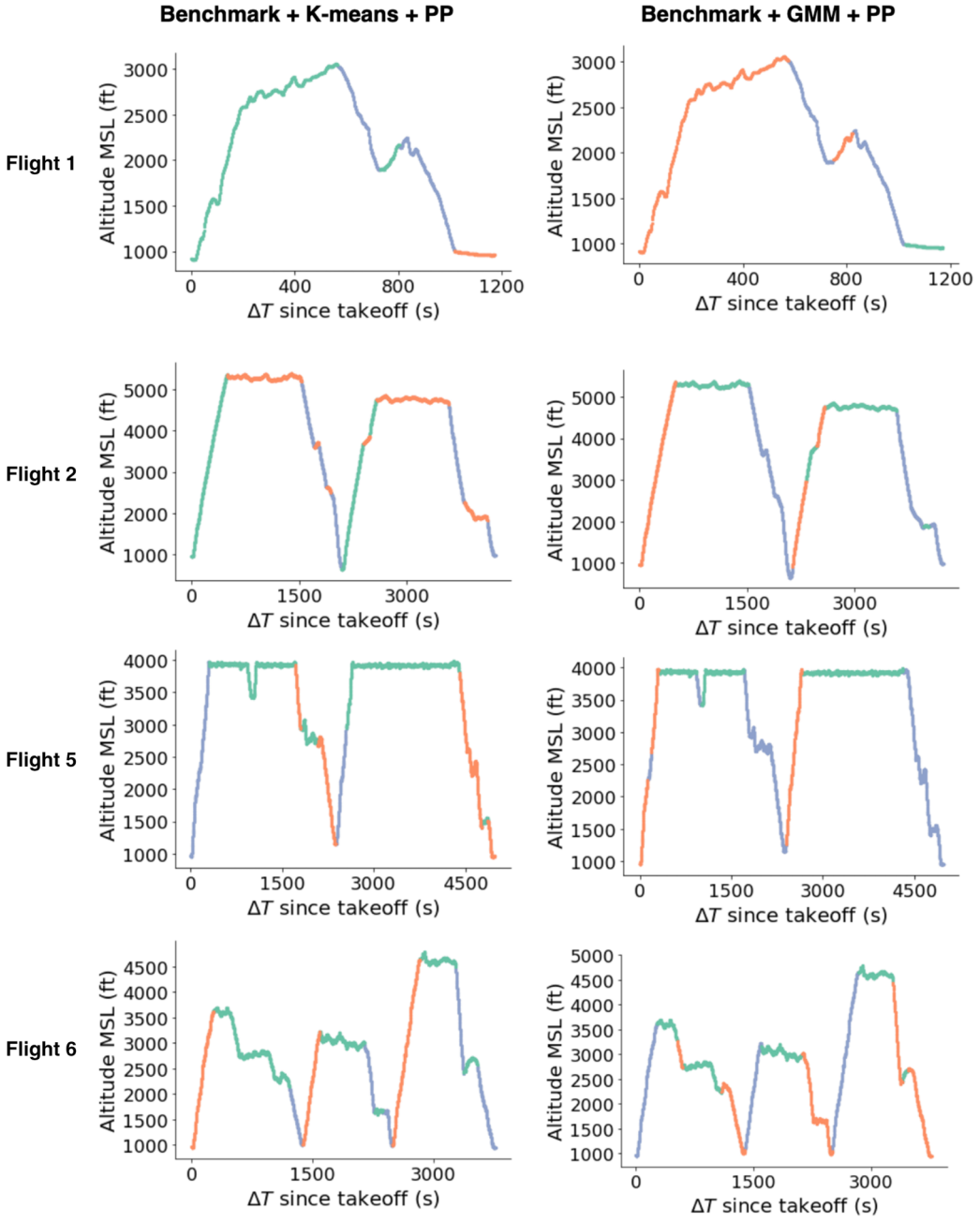

Fig. 8 Clustering results for the four test flights of a C-172 aircraft.

Table 9 Total scoring based on qualitative criteria for all the C-172 test flights

	Criterion	Benchmark+ K-means + PP	Benchmark+ GMM + PP
Flight 1	A: Climb	√	√
	B: Cruise	√	√
	C: Descent	√	√
	D: Transient	∼	∼
	E: Steps	√	√
Flight 2	A: Climb	∼	∼
	B: Cruise	∼	∼
	C: Descent	√	√
	D: Transient	∼	√
	E: Steps	∼	√
Flight 5	A: Climb	×	∼
	B: Cruise	∼	∼
	C: Descent	√	∼
	D: Transient	∼	∼
	E: Steps	√	√
Flight 6	A: Climb	√	∼
	B: Cruise	×	∼
	C: Descent	×	√
	D: Transient	√	∼
	E: Steps	√	∼
Total scoring out of 20		13.5	14.5

stepped descents in between or after. Both methods result in adequate quality clustering, with some obvious mislabeling. The GMM incorrectly clusters part of the second climb as cruise, and the K-means incorrectly groups the last cruise with parts of the previous descent. These mislabeling, however, would be easy to spot and correct by a flight instructor who relies on these tools. Flight 5 is also more challenging to cluster than previous flights: the cruise phase does not use an autopilot, and so there is noise in the data; they include a sudden descent and climb back to the same altitude, and lastly a long and nonstable descent. The GMM provides a preferable result, with the exception of a small segment at climb that is classified as descent. The K-means mislabels the steep descent, which is undesirable. Flight 6 includes multiple cruises with step descents at different altitudes. Neither method classifies the flight entirely correctly. The K-means is worse because it does not distinguish well between any cruise/descents reliably, whereas the GMM only mislabels one of those at 1500 ft. The scoring for the C-172 flights is shown in Table 9.

Overall, the qualitative evaluation of benchmark + GMM + PP (score = 32) and benchmark + K-means + PP (score = 32.5) indicates that both methods are adequately identifying the phases of flight in various scenarios. The assumptions of the GMM are a good fit for clustering flight data (i.e., a Gaussian mixture is a good approximation for the flight data distributions within the flight phases we studied). The established dimensionality reduction techniques do not offer any benefit on feature selection over a feature selection that omits engine-specific information.

VII. Future Research

In future research, we plan to consider the time dimension, which we did not take into consideration for this work. Time-series models and processing may be helpful at improving the quality of clustering because they allow us to take into consideration the history (and phase) of previous data in determining the next phase. We also plan to investigate other types of flights that are not cross country, and we will likely have more segments that are not well defined (e.g., maneuver and student pilot training flights); we will also expand on other flight phases (e.g., approach, takeoff, and landing).

VIII. Conclusions

Human intervention in the task of flight phase identification can introduce personal biases in training algorithms and decision-making. In this work, the performance of various combinations of unsupervised machine learning methods was investigated for the task of flight clustering from data records after feature selection was investigated. Four different methods of feature selection were studied: the low variance filter, the high correlation filter, principal component analysis, and autoencoders; it was found that these methods do not offer any benefit or advantage for the task of flight phase identification as compared to the benchmark that simply removes engine-specific features. Then, the benchmark features were used and compared to the clustering result of two algorithms: K-means and Gaussian mixture models. These methods were tested for eight cross-country test flights in total, from SR-20 and C-172 aircraft. A postprocessing method was developed for the flight phase identification to correct the labels for some small segments that were mislabeled. Overall, the qualitative evaluation indicates that both methods are adequate at identifying the phases of flight in various scenarios.

The main advantage of the algorithms used is their ability to identify and cluster phases of flight well without any human intervention. Some of the issues seen in the training data (e.g., in flight 7 of Fig. 7, where the K-means algorithm identified three steps in the descent and GMM treated it as a straight descent) are issues that even subject matter experts would have faced. There is no ground truth to the problem tackled. Labeling flights to use to train and evaluate algorithms would only estimate the ground truth; as a result, the algorithm would treat the step descent as individual steps or as a straight descent, depending on the opinion of the subject matter expert who labeled the flights.

Acknowledgments

The authors acknowledge the Oklahoma State University Flight Center for providing the Garmin G1000 data used in this research. The authors acknowledge the Tufts University High Performance Computing Cluster, which was used for training the autoencoder models. The authors would like to also thank the reviewers for their comments that greatly improved the paper. The first two authors contributed equally to this work.

References

[1] Zhang, Q., Mott, J. H., Johnson, M. E., and Springer, J. A., "Development of a Reliable Method for General Aviation Flight Phase Identification," *IEEE Transactions on Intelligent Transportation Systems*, Vol. 23, No. 8, 2021, pp. 11,729–11,738. https://doi.org/10.1109/TITS.2021.3106774

[2] Chatterji, G. B., "Fuel Burn Estimation Using Real Track Data," *11th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference, Including the AIAA Balloon Systems Conference and 19th AIAA Lighter-Than-Air Conference*, AIAA Paper 2011-6881, 2011. https://doi.org/10.2514/6.2011-6881

[3] Chati, Y. S., and Balakrishnan, H., "Analysis of Aircraft Fuel Burn and Emissions in the landing and Take Off Cycle Using Operational Data," *International Conference on Research in Air Transportation*, 2014, https://www.icrat.org/previous-conferences/6th-international-conference/ICRAT2014_PROGRAM_digital_v1.pdf.

[4] Fala, N., and Marais, K., "Detecting Safety Events During Approach in General Aviation Operations," *16th AIAA Aviation Technology, Integration, and Operations Conference*, AIAA Paper 2016-3914, 2016. https://doi.org/10.2514/6.2016-3914

[5] "Phase of Flight Definition and Usage Notes (1.3)," Commercial Aviation Safety Team/International Civil Aviation Organization Common Taxonomy Team, 2013.

[6] "Review of US Civil Aviation Accidents—Calendar Year 2011," National Transportation Safety Board Annual Review ARA-14/01, Washington, D.C., 2014.

[7] Huang, C., and Johnson, M., "Fuel Flow Rate and Duration of General Aviation Landing and Takeoff Cycle," *16th AIAA Aviation Technology, Integration, and Operations Conference*, AIAA Paper 2016-4366, 2016. https://doi.org/10.2514/6.2016-4366

[8] Goblet, V., Fala, N., and Marais, K., "Identifying Phases of Flight in General Aviation Operations," *15th AIAA Aviation Technology, Integration, and Operations Conference*, AIAA Paper 2015-2851, 2015. https://doi.org/10.2514/6.2015-2851

[9] Kuzmenko, N., Ostroumov, I., Bezkorovainyi, Y., Averyanova, Y., Larin, V., Sushchenko, O., Zaliskyi, M., and Solomentsev, O., "Airplane Flight Phase Identification Using Maximum Posterior Probability Method," *2022 IEEE 3rd International Conference on System Analysis & Intelligent Computing (SAIC)*, IEEE, New York, 2022, pp. 1–5. https://doi.org/10.1109/SAIC57818.2022.9922913

[10] Zhang, Q., and Mott, J. H., "Improved Framework for Classification of Flight Phases of General Aviation Aircraft," *Transportation Research Record*, Vol. 2677, No. 3, 2023, pp. 1665–1675. https://doi.org/10.1177/03611981221127016

[11] Chati, Y. S., and Balakrishnan, H., "Aircraft Engine Performance Study Using Flight Data Recorder Archives," *2013 Aviation Technology, Integration, and Operations Conference*, AIAA Paper 2013-4414, 2013. https://doi.org/10.2514/6.2013-4414

[12] Gong, C., and McNally, D., "A Methodology for Automated Trajectory Prediction Analysis," *AIAA Guidance, Navigation, and Control Conference and Exhibit*, AIAA Paper 2004-4788, 2004. https://doi.org/10.2514/6.2004-4788

[13] Paglione, M., and Oaks, R., "Determination of Horizontal and Vertical Phase of Flight in Recorded Air Traffic Data," *AIAA Guidance, Navigation, and Control Conference and Exhibit*, AIAA Paper 2006-6772, 2006. https://doi.org/10.2514/6.2006-6772

[14] Kelly, W. E., III, and Painter, J. H., "Flight Segment Identification as a Basis for Pilot Advisory Systems," *Journal of Aircraft*, Vol. 43, No. 6, 2006, pp. 1628–1635. https://doi.org/10.2514/1.20484

[15] Sun, J., Ellerbroek, J., and Hoekstra, J., "Large-Scale Flight Phase Identification from ADS-B Data Using Machine Learning Methods," *7th International Conference on Research in Air Transportation*, ICRAT, Philadelphia, PA, June 2016.

[16] Sun, J., Ellerbroek, J., and Hoekstra, J., "Flight Extraction and Phase Identification for Large Automatic Dependent Surveillance–Broadcast Datasets," *Journal of Aerospace Information Systems*, Vol. 14, No. 10, 2017, pp. 566–572. https://doi.org/10.2514/1.I010520

[17] Kovarik, S., Doherty, L., Korah, K., Mulligan, B., Rasool, G., Mehta, Y., Bhavsar, P., and Paglione, M., "Comparative Analysis of Machine Learning and Statistical Methods for Aircraft Phase of Flight Prediction," *International Conference on Research in Air Transportation 2020, 9th International Conference*, Sept. 2020, https://www.icrat.org/previous-conferences/9th-international-conference/papers/.

[18] Tian, F., Cheng, X., Meng, G., and Xu, Y., "Research on Flight Phase Division Based on Decision Tree Classifier," *2017 2nd IEEE International Conference on Computational Intelligence and Applications (ICCIA)*, IEEE, New York, 2017, pp. 372–375. https://doi.org/10.1109/CIAPP.2017.8167242

[19] Arts, E., Kamtsiuris, A., Meyer, H., Raddatz, F., Peters, A., and Wermter, S., "Trajectory Based Flight Phase Identification with Machine Learning for Digital Twins," German Soc. for Aeronautics and Astronautics (DGLR), 2022. https://doi.org/10.25967/550191

[20] Chin, H.-J., Payan, A., Johnson, C., and Mavris, D. N., "Phases of Flight Identification for Rotorcraft Operations," *AIAA SciTech 2019 Forum*, AIAA Paper 2019-0139, 2019. https://doi.org/10.2514/6.2019-0139

[21] Liu, D., Xiao, N., Zhang, Y., and Peng, X., "Unsupervised Flight Phase Recognition with Flight Data Clustering Based on GMM," *2020 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, IEEE, New York, 2020, pp. 1–6. https://doi.org/10.1109/I2MTC43012.2020.9128596

[22] Bank, D., Koenigstein, N., and Giryes, R., "Autoencoders," Preprint, submitted 12 March 2020, https://arxiv.org/abs/2003.05991. https://doi.org/10.48550/arXiv.2003.05991

[23] Klambauer, G., Unterthiner, T., Mayr, A., and Hochreiter, S., "Self-Normalizing Neural Networks," *Advances in Neural Information Processing Systems*, edited by I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Vol. 30, Curran Assoc., Red Hook, NY, 2017, https://proceedings.neurips.cc/paper/2017/file/5d44ee6f2c3f71b73125876103c8f6c4-Paper.pdf [retrieved 25 July 2023].

[24] Xu, D., and Tian, Y., "A Comprehensive Survey of Clustering Algorithms," *Annals of Data Science*, Vol. 2, No. 2, 2015, pp. 165–193. https://doi.org/10.1007/s40745-015-0040-1

[25] von Luxburg, U., Williamson, R. C., and Guyon, I., "Clustering: Science or Art?" *Proceedings of ICML Workshop on Unsupervised and Transfer Learning, Proceedings of Machine Learning Research*, edited by I. Guyon, G. Dror, V. Lemaire, G. Taylor, and D. Silver, Vol. 27, MLR Press, Bellevue, WA, 2012, pp. 65–79.

[26] Carreira-Perpinán, M. A., and Williams, C. K., "On the Number of Modes of a Gaussian Mixture," *International Conference on Scale-Space Theories in Computer Vision*, Springer, Berlin, 2003, pp. 625–640. https://doi.org/10.1007/3-540-44935-3_44

[27] Ahmed, M., Seraj, R., and Islam, S. M. S., "The k-Means Algorithm: A Comprehensive Survey and Performance Evaluation," *Electronics*, Vol. 9, No. 8, 2020, Paper 1295. https://doi.org/10.3390/electronics9081295

[28] Moon, T., "The Expectation-Maximization Algorithm," *IEEE Signal Processing Magazine*, Vol. 13, No. 6, 1996, pp. 47–60. https://doi.org/10.1109/79.543975

H. Oh
Associate Editor