A Machine Learning Framework to Predict General Aviation Traffic Counts

A Case Study for Nice Cote D'Azur Terminal Control Center

Amir Abecassis
OPTIM team - ENAC lab
French Civil Aviation University (ENAC)
Toulouse, France
amir.abecassis@enac.fr

Daniel Delahaye
OPTIM team - ENAC lab
French Civil Aviation University (ENAC)
Toulouse, France
daniel.delahaye@enac.fr

Moshe Idan
Faculty of Aerospace Engineering
Technion-Israel Institute of Technology
Haifa, Israel
moshe.idan@technion.ac.il

***Abstract*—General Aviation traffic prediction is a major concern for Air Navigation Service Providers as it has a direct impact on air traffic flow and capacity management measures. However, today, few tools are available to address this issue. This paper proposes a methodology to predict GA traffic based on Machine Learning models training with historical data. Initial promising results are obtained on Nice Cote D'Azur Terminal Control Center sectors case study using meteorological and calendar data with an increase of the prediction performance of 25% compared to current tools used in operation.**

***Keywords*—Machine Learning, Demand and Capacity balancing, General Aviation, Air Traffic Control, Entry Counts**

I. Introduction

A. Presentation of the operational problem

General Aviation (GA) operations prediction is a main issue for all Air Traffic Control (ATC) stakeholders as a key point for improving safety performance and operations. Indeed, flights plan submission is often not mandatory for these flights. Moreover, flight plan submission, if occurs, can be done very close to the flight operation (e.g. 15 minutes before take-off.)

Less than 5% of GA flights fly under flight plans in France. This entails a major limitation for ATS (Air Traffic Services) operation preparation. Many major French airports are subject to both GA and Commercial Aviation, where Commercial Aviation framework is clearly defined while GA operations can occur anytime when the airport is open. This forces DSNA (French Air Navigation Service) to organize its service taking this major difference into account to guarantee safety.

This dictates the need to always have staff available to handle both types of traffic, knowing that in case of emergency or unexpected events, GA traffic can have an impact on Commercial Aviation traffic, which, opposed to GA, can be safely regulated. With this in mind, we demonstrate that predicting GA traffic is of primary importance for these airports, not only for GA but for all traffic operations. For example, in Nice airport, the third largest in France in terms of traffic, GA represents a significant portion of the traffic in the Nice Terminal Control Center (TCC): each day, on average, there are as many GA flights flying through the TCC sectors as Commercial flight departures/arrivals in the Nice airport. In the peak season, it represents more than 500 GA flights per day; GA traffic in 2022 has exceeded the 2019 level.

Therefore, the claim is that a GA traffic prediction tool in terminal sectors is needed to have a complete and reliable Air Traffic Flow and Capacity Management (ATFCM) process that will increase ATC performance.

B. DSNA and Nice TCC context

The services delivered by DSNA to GA flights include, at least, the Flight Information Service and Alerts, which are mandatory as long as GA flights contact the ATS by radio in class E to class G airspace. For class C or D airspace (no class B in France), the Control Service is also mandatory for these flights. DSNA has set up SIV (Secteur d'Information de Vol - Flight Information Sectors) managed by controllers (ATCO) to deliver these services to GA flights in France airspace, level of which depends on the class of airspace in each SIV sector. In France, those ATCO are mostly located in airports.

For instance, as presented in Fig. 1, Nice TCC is currently composed of three SIVs that geographically divide a big part of TCC airspace under Flight Level (FL) 195: SIV1 (grouping SIV1.1 and 1.2 up to FL175 and FL145, respectively), SIV2 (until FL145) and SIV3 (grouping SIV3.1 and 3.2 until FL115 and FL145, respectively). These SIVs can contain Control Traffic Regions (CTR) and Terminal Manoeuvring Areas (TMA) sectors that manage operations to and from a specific aerodrome. However, Nice SIV sectors do not contain only this kind of terminal sectors. The management of this intricate airspace scheme is further complexified by the shifting nature of the traffic, with significant changes depending on the season, day of the week, weather, or special events.

Consequently, specific controllers are entirely dedicated to GA flights and related services in the SIV sectors. Today, on average, 20% of Nice TCC controllers oversee these flights in these SIV sectors (INFO positions), the actual number fluctuating accordingly with the level of GA traffic. This highlights the importance of having a reliable tool to predict the GA traffic.

Today, DSNA does not manage GA traffic globally, since it does not have the same impact on all its airports. Nice TCC

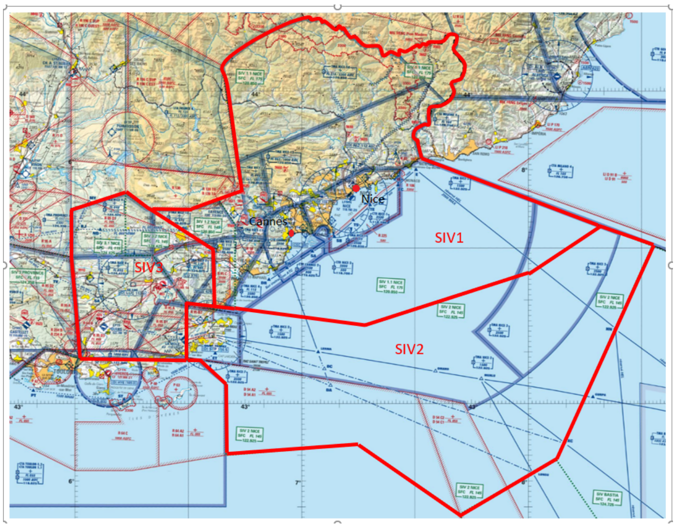

Fig. 1. SIV sectors organization in Nice TCC airspace.

is one of the most impacted ones, as detailed above. A basic GA traffic prediction model, developed and assessed locally using past data, was found to provide insufficient accuracy. It is presented in section III-C3, as it will serve as a reference baseline for the models derived in this study.

The objective of this paper is to present a methodology to predict GA traffic many hours in advance for ATFCM purpose. As a first step, the prediction horizon was chosen as four hours. The proposed tool is based on a data driven approach by means of Machine Learning (ML), evaluated in the context of the Nice SIV. It can be easily adapted to other airports or sectors of interest. The remaining of the paper is structured as follows. We discuss the previous literature on ML applied to ATM tasks, specifically for trajectory prediction and Demand and Capacity Balancing (DCB), then for general aviation prediction in section II. Section III describes the global ML methodology proposed to address the problem of entry counts prediction in SIV sectors. In section IV, we present and analyze the performance of the resulting solution. Finally, concluding remarks and future work directions are provided in section V.

II. State of the Art

First, we consider how ML has been used in Air Traffic Management (ATM). Prediction of ATM features thanks to ML has been widely studied over the past years. One of the most recurrent topics in ML applied to ATM is 4-Dimensions (4D) trajectory prediction for Commercial and GA flights.

The DART SESAR (Single European Sky ATM Research) project [1] has worked intensively on this topic. The first objective was to evaluate ML methods to increase reliability of 4D trajectory prediction, in order to build delay-reducing ATFCM mechanisms in a Trajectory Based Operations context. The project has proposed a hybrid method for single 4D trajectory based on unsupervised learning (clustering techniques) and supervised learning (many methods were considered: e.g., Multi-Layer Perceptron Neural Network (ANN) [2, Ch. 6], Hidden Markov Models [3], Linear Regressors [4, Ch. 4], Classification and Regression Trees models [4, Ch. 6]). It has demonstrated their efficiency in the En-Route airspace context, but with limitations in TMA. The reached objective was mainly linked to the precision of the input data provided to the models, in particular flight plans.

Regarding the TMA, the study in [5] has implemented a two levels approach: first, an unsupervised ML layer was used to clean and pre-process data (based on K-Means clustering and Density-Based spatial clustering of applications with noise algorithms [4, Ch. 9]), followed by a supervised ML layer with ANN to predict Estimated Time of Arrival (ETA). More globally, [6] presents a general review on trajectory prediction research over past decade.

As an extension, a lot of work has been dedicated to predicting specific parts of the trajectory and not the whole 4D trajectory itself. The method in [7] predicts the total length of descent flight, starting from mode S radar data with supervised ML methods as ANN, Gradient Boosting Decision Trees (GBDT), and demonstrates better performance than BADA (Base of Aircraft DAta) baseline. Ramon Dalmau et. al. [8] have implemented a ML model based on GBDT to improve take-off time predictions. In addition, [9] focused on Standard Instrument Departure (SID) and Standard Instrument Arrival (STAR) prediction many hours in advance with the use of a GBDT model based on time, trajectory and meteorological data. In general, it can be stated that 4D trajectory prediction has been mainly studied for Commercial Aviation.

Some studies, however, were dedicated to GA. A Recurrent Neural Network (RNN) based on Long Short Term Memory (LSTM) layers to predict GA aircraft flight trajectory based on ADSB (Automatic dependent surveillance-broadcast) data was proposed in [10]. It could be used as a workaround in case of a very short breakdown of the surveillance chain of an ATM system. It can be concluded that the medium-term prediction (some hours before) ML models are often possible thanks to the existence of Flight Plan information provided by the Network Manager. If not available, the prediction can only be very short term (some minutes). In the GA context, SESAR WP-E Research ProGA, that investigated the concept of providing pilots of light GA aircraft with an electronic representation of the estimated location of surrounding traffic several minutes ahead of time, also presented the challenge of medium-term and long-term GA flights location prediction without any flight plans information [11].

DCB was also a major field studied using ML tools. [12] presents a two-step method with multinomial logistic regression to estimate the number of controller positions to open depending of the level of expected traffic and a tree search algorithm to identify the best sector configuration to deploy during the period to support Air Control Center (ACC) occupancy counts (OCC: number of flights in a sector during a selected time period). [13] and [14] use 4D trajectory and meteorological data to predict En Route ACC sector OCC by training many types of models (GBDT Regressor, Random Forest (RF) Regressor, Support Vector Machine Regressor, ANN, RNN with LSTM layers, and more). Similarly, [15]

sesar JOINT UNDERTAKING
12th SESAR Innovation Days
5-8 December 2022, Budapest
HungaroControl
sesar DIGITAL ACADEMY

targets to predict En Route sector OCC considering features linked to sector complexity and flows characteristics inside the sector using a RF Regressor. In addition, [16] predicts the number of delayed flights and ATFM delay using many different ML techniques by proposing a model based on a deep Convolutional NN allowing to extract characteristics of regulation data (identifying spatial and temporal correlations). Finally, [17] presents a probabilistic model to estimate OCC for En Route sectors in the frame of the SESAR COPTRA project. Once again, research on DCB was widely dedicated to commercial aviation as, by nature, a reliable medium-term flight trajectory prediction is based on inputs available for commercial flights. For GA flights, that are less constrained, DCB target is more complex to reach.

We now focus on studies dedicated to GA operations. As already stated, [11] is dedicated to GA flight trajectory prediction and, to the best of our knowledge, seems to be the closest, in ATM context, to the problem addressed in the current study. They outline the limits of stochastic filters that use the Bayes' theorem to predict flight dynamic, based on its current position, for medium to long term predictions, as long-term position of the flight is more linked to the pilot intent than to the current flight position. It is important to note that pilot intent can evolve with time (before and during the flight) and is strongly linked to the regional context.

Looking outside of ATC area but still related to Air Transportation, research on GA has been active on FAA (Federal Aviation Administration) side for airport strategic operations management. This is often limited to statistical studies and models that are proposed, e.g., [18] and [19], to forecast airport-level GA demand. Regression models are used to estimate the total number of GA operations in a year at a given airport. The study in [20] proposes an approach based on Linear Regression, for estimating the percentage of GA operations per year in an airport. Strategic operations management can rely on a macroscopic prediction approach as a large time horizon is considered (some years) and the aim is more to identify a trend than a precise value. Finally, [21] is the closest study to the problem addressed in this paper, as it tries to count flights per day in a non-towered airport. The approach is very simple and based on very few data. Moreover, its scope is limited because it does not differentiate between commercial and GA flights.

As a matter of fact, even if these studies are (even partially) related to GA operations, the problems addressed are different: very high level methods are used (input data, models) and very large time horizons are considered. Therefore, their results cannot be used as such: our GA ATC problem, that intends to predict some days to some hours before operations, has not been addressed in the literature to the best of our knowledge.

III. Methodology

This section presents our methodology on data and model to tackle the operational problem introduced above.

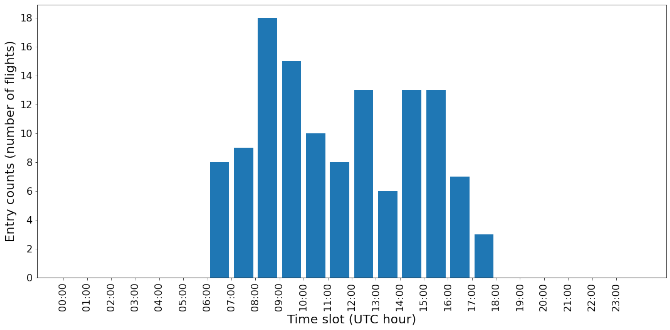

Fig. 2. Entry counts in SIV1 on Sunday, August 22, 2021.

A. Data presentation and preparation

As reported in [11], GA traffic strongly depends on pilot intent that is hard to predict using limited data. Consequently, we consider many different data sources to tackle GA traffic prediction for which it should be possible to extract the pilot intent information from regional/contextual data. First, the main source of input is the recorded surveillance data provided by DSNA. For each month of the considered period, operational experts have provided complete recorded traffic by Nice TCC ATM surveillance system. This encompasses both commercial and GA traffic known by the system.

1) Surveillance data: A first step was to extract and analyze the traffic occurring in each SIV sector. The output of this step was a list of recorded flights with their characteristics for each SIV, including time of entry and time of exit for each crossed sector. Data analysis allowed to ensure data completeness and consistency. These data were sorted to extract GA traffic, performed with Nice operational staff to identify flight families depending on their Mode A code. This allowed to remove commercial flights and identify various families of GA flights. In addition, some specific non-commercial flights, such as military, fire-fighting, hospital, helicopters, were assumed to be out of the scope of this study and thus were removed, resulting in a list of relevant GA flights per sector with related characteristics.

The second step was to construct the SIV entry counts. In this study, we considered the number of flights in a time period of one hour for a total of 24 time slots per day. The flights where sorted into these time slots according to their entry time into a sector. The output is a time series per SIV (with time slot as timestamp and entry counts in the sector) over the considered period. Figure 2 presents an example of counts on August 22, 2021. Note that these counts will be used as the prediction target of the models derived in this study.

2) Meteorology data: Two types of meteorology data have been processed in this study: METAR (Meteorological Aerodrome Report) and TAF (Terminal Aerodrome Forecast). They are, respectively, meteorology observation and forecast data in an aerodrome. METAR is provided each hour (or thirty minutes) for each equipped aerodrome. TAF are provided until 30 hours in advance and provide the meteorology forecast of the next few hours. TAF and METAR data are available in free text format standardized by OACI. Sources used

TABLE I
METAR AND TAF KEYWORDS EVALUATION.

Keyword	Type	Level
FEW	Clouds	1
NSC / CAVOK / NOSIG	Visibility	1
FU / DZ / MIFG	Significant weather	2
$\leq$ SCT080	Clouds	2
$\geq$ G20 and $<$ G30	Gust	3
CB / TCU	Clouds	3
$>$ BKN030 and $\leq$ BKN080	Clouds	3
SA / DU / BCFG / VCFG / VCSH	Significant weather	3
$>$ OVC080	Clouds	3
$\leq$ OVC080	Clouds	4
M10	Temperature	4
$\geq$ 20KT and $<$ 30KT	Wind	4
$\geq$ G30 and $<$ G40	Gust	4
BR / HZ / RA / SN	Significant weather	4
$>$ BKN015 and $\leq$ BKN030	Clouds	4
RASN / RADZ / DZRA / DZSN / VCTS	Significant weather	4
PO / TS / SHRA / SHSN / GR / PL	Significant weather	5
VA / FG / FC / SQ / DS / SS / GS / IC	Significant weather	5
$\geq$ 30KT	Wind	5
$\geq$ G40	Gust	5
$\leq$ BKN015	Clouds	5
”1500”	Visibility	5
SHGS / SHGSRA / SHRASN / VV///	Significant weather	5

for METAR relevant for Nice sectors were extracted from historical METAR of Iowa Environmental Mesonet from Iowa State University [22]. TAF were extracted from OGIMET [23]. It is important to note that it was difficult to obtain TAF data as very few sites are recording them.

The operational justification to use both sources is that METAR deals with meteorology observation. Consequently, METAR related to the predicted time slot H is not available when performing entry counts prediction few hours in advance. The pilot has the same issue when preparing his flight. To decide to fly or not, he must rely on the following incomplete information:

- METAR of the previous hours for the slot H, as they are assumed to be strongly correlated;
- TAF that provides target slot H meteorology forecast.

The following step was to use METAR and TAF data to extract their impact on air traffic operations. The rationale was to use the expertise of the operational personnel (controllers, pilots) to define a simple scaling scheme that, using the various keywords inside METAR and TAF messages, produces a 1-5 rating of the meteorology, where higher values imply a more critical weather. Each METAR and TAF keyword is evaluated as defined in Table I. The final and global METAR (or TAF) evaluation considers the maximal value from all message keywords. It should be noted that features presented in this table are union of features available in METAR and TAF, while some of them (e.g. NSC/CAVOK) are relevant only to METAR. Consequently, specific pipelines were implemented to evaluate the meteorology for each time slot according to information available four hours before the specific time slot to predict: METAR of the time slot $H-4$ and the latest TAF available (2AM UTC).

3) Calendar data: The last category is related to calendar data. For each day, its date is converted to the week number and the day number in the week. Note that previous evaluations led to discard the month number as the input feature due to its lesser effectiveness on training and validation sets. Then, data is enriched with time slot of the day, information about whether the day is a holiday or a bank holiday, and information about whether a specific event is organized during the day. For this last feature, a list of events impacting regional GA traffic according to ATC experience has been identified and is presented in Table II.

TABLE II
LIST OF SPECIAL EVENTS AROUND NICE IN 2019.

Event name	Start date in 2019	End date in 2019
Monte Carlo Rallye	24th of Jan	27th of Jan
Nice Carnival	16th of Feb	2nd of March
Fête des Citrons Menton	16th of Feb	3rd of March
MIPIM	12th of March	15th of March
Cannes Festival	14th of May	25th of May
Monaco GP	26th of May	26th of May
Voiles d’Antibes	5th of June	9th of June
Top Marques	30th of May	3rd of June
Yacht Festival Cannes	10th of Sept	15th of Sept
Régates Royales Cannes	22th of Sept	29th of Sept
Monaco Yacht Show	25th of Sept	28 of Sept

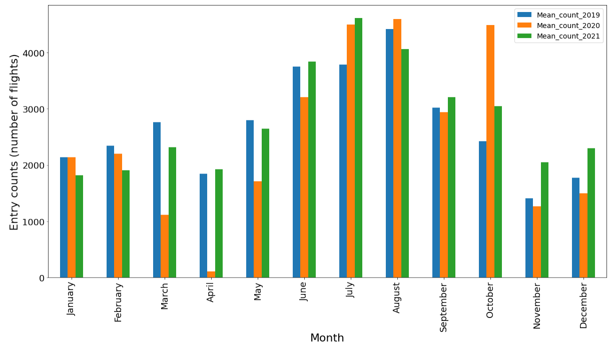

Fig. 3. Entry counts mean evolution on SIV1 over 2019 to 2021 period.

Finally, the scope of historical data to be used by our model had to be defined. DSNA has provided data from beginning of 2019 to end of May 2022. However, COVID-19 period had important impacts on GA traffic over a part of this period. The large disparity over March to May period of each year is shown on Fig. 3. Consequently, at this stage, the dataset was limited to 2019, 2021 and 2022 period, removing 2020 period to avoid mainly the untypical impact of COVID-19 (lockdowns, curfews). A smoother approach could be possible in future work performing a deep analysis of COVID-19 impact over the period.

B. Model design

1) General representation: In this section, we present our ML model. The goal is to predict a variable y (our target), i.e., GA entry counts in a SIV defined as an integer (extracted from our surveillance data section III-A1), from a set of explanatory features x, for which x and y are respectively drawn from probability distributions X and Y. Considering m features, x is a vector $\in \mathbb{R}^m$ and $y \in \mathbb{R}$ (instead of $\mathbb{N}$ to simplify the problem). The random variable X has a probability density function (pdf) $p_X(x)$. Our data presented in the previous section is defined by the dataset D, with $D = (x_1, y_1), ..., (x_N, y_N) \subseteq (X, Y)$. Consequently, the problem can be addressed by a supervised ML regression technique, which purpose is to approximate a function f such that: $y = f(x)\ \forall\ (x, y) \in (X, Y)$, considering $f(x)$ as the prediction of our model and y as the true value.

Two parameters have to be defined to solve this problem. First F, our hypothesis space of functions, $(f \in F)$, has to be defined. Then, a loss function $L : Y^2 \rightarrow \mathbb{R}$ is needed to measure $\forall(x, y) \in (X, Y)$ how well f is fitting (x, y). $\forall(x, y) \in (X, Y)$, the cost is modelled by $L[f(x), y]$. The function f is found by minimizing a risk function defined as the expectation of the cost taken on the whole joint probability distribution (X, Y), i.e.,

$$\begin{aligned} R(f) &= E_{X,Y}\left[L(f(X), Y)\right] \\ &= \int_{(X,Y)} L\left[f(x), y\right] p_{X,Y}(x, y) dx dy, \end{aligned} \tag{1}$$

where $p_{X,Y}$ is the pdf of the joint probability (X, Y).

This problem is intractable as the joint distribution (X, Y) is unknown and our knowledge is limited to D. To yield a tractable optimization problem, it is assumed that $p_{X,Y}$ of (X, Y) can be approximated by an empirical distribution $\hat{p}(x, y)$, defined by the training set that will be used to train the model (part of D). In addition, it is assumed that the data in D are independent and identically distributed (i.i.d). With these assumptions, the empirical risk $R_e(f, D)$, defined by

$$\begin{aligned} R_e(f, D) &= E_{x,y\sim\hat{p}}\left[L(f(X), Y)\right] \\ &= \frac{1}{|D|}\sum_{i=1}^{N} L\left[f(x_i), y_i\right], \end{aligned} \tag{2}$$

converges to $R(f)$ of (1) as $|D|$ increases. Hence, our ML problem is restated as

$$f^* = \underset{f \in F}{\operatorname{argmin}}\ R_e(f, D). \tag{3}$$

The goal of this model, even if trained on a part of D, is to generalize well on an unseen dataset denoted by D_{test}, that is also assumed to be i.i.d and sampled from the same (X, Y) distribution. Unfortunately, D is a finite and limited set of data. Therefore, there is no guarantee that R_e converges to R. Consequently, our model can be prone to over-fitting. Specifically, if the function space F is large, access to complex functions can yield an exact fit of a finite training set. Without setting any limit on F, this may causing generalization difficulties: the fact that the empirical risk is very low does not guarantee that so is the actual risk. To address over-fitting, F has to be chosen carefully by balancing prediction accuracy (bias) and prediction error variance. It will also affect the training methodology, as is addressed in the following sections. One method to control F is to use regularization hyper-parameters in ML models limiting the set of possible functions [2].

TABLE III
LIST OF FEATURES.

Feature name	Short Description	Type
Meteorology		
N_MET$_{H\text{-}4}$	Nice past METAR	Numerical
C_MET$_{H\text{-}4}$	Cannes past METAR	Numerical
N_TAF$_H$	Nice TAF	Numerical
C_TAF$_H$	Cannes TAF	Numerical
N_TAF$_J$	Nice average day TAF	Numerical
C_TAF$_J$	Cannes average day TAF	Numerical
Calendar		
H	Time slot	Numerical
DoW	Day of the week	Numerical
WN	Number of the week	Numerical
S$_{\text{hol}}$	Holiday	Boolean
S$_{\text{ev}}$	Special Event	Boolean

2) Definition of input features: The input features, previously denoted by x, has been defined carefully considering operational expertise and the correlation of those features with our target variable y, i.e., the predicted number of GA flights. It should be noted that the features list presented in this section and in Table III is tailored for SIV1. It allows to present SIV1 related results in section IV.

N_MET$_{H\text{-}4}$ and C_MET$_{H\text{-}4}$ are the meteorology information scores (see section III-A2) extracted from Nice and Cannes METAR for the time slot of 4 hours before the target slot (called slot H) to predict. N_TAF$_H$ and C_TAF$_H$ are TAF scale scores of the meteorology forecast for target time slot H, extracted from Nice and Cannes TAF which is available at 2AM UTC the same day. N_TAF$_J$ and C_TAF$_J$ are full day meteorology forecast indicators for the day that contains the target time slot H for Nice and Cannes. If, for a specific day, N_TAF$_{Hi}$ and C_TAF$_{Hi}$ are meteorology forecast for time slots $H_i, i = 1, \ldots, 24$ in Nice and Cannes, then

$$\text{N\_TAF}_J = \frac{1}{24}\sum_{i=1}^{24} \text{N\_TAF}_{Hi}, \tag{4a}$$

$$\text{C\_TAF}_J = \frac{1}{24}\sum_{i=1}^{24} \text{C\_TAF}_{Hi}. \tag{4b}$$

Nice and Cannes aerodrome meteorology have been chosen for SIV1 but it is important to note that this is not a perfect choice. Indeed, Fig. 1 shows that SIV1 geography covers the north of the region and Nice and Cannes aerodrome meteorology is not fully representative of this large sector: the north of the sector covers a mountainous terrain that can have very specific weather characteristics but there is no meteorological station covering this area. All these features are numerical as built from our meteorology scale.

The remaining set of features are related to calendar data. H and DoW are numerical features defining the time slot of the day (24 possible values, for instance 10 means 10AM to 11AM UTC) and day of the week (Monday to Sunday, encoded with numerical value from 0 to 6). Following a statistical analysis on the 2019-2021 period, the dataset for H was limited to values between 6 to 19, which means considering GA traffic between 6AM to 8PM UTC. Indeed, Fig. 4 presents the mean

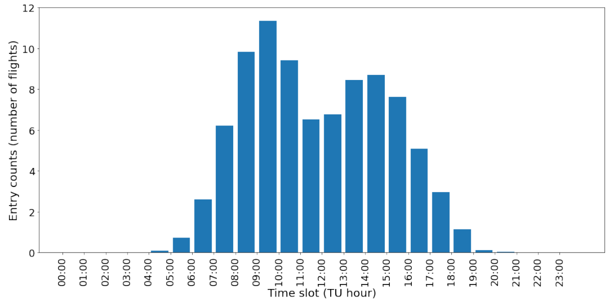

Fig. 4. Mean entry counts on SIV1 over 2019 to 2021 period.

GA traffic in SIV1 per time slot over the 2019 to 2021 period and shows that the 6AM to 8PM UTC period contains 99% of the daily traffic.

WN is related to week number in a year (Week 1 to 53). This feature has been converted to keep periodicity information, as W1 and W53 are close from a time and seasonal perspective but are greatly separated numerically. After a review of possible solutions, this single feature was converted into two features using cyclical features encoding to keep seasonality information [24], i.e.,

$$\mathrm{WN}^{\cos} = \cos\left(\frac{2\pi \cdot \mathrm{WN}}{53}\right), \tag{5a}$$

$$\mathrm{WN}^{\sin} = \sin\left(\frac{2\pi \cdot \mathrm{WN}}{53}\right). \tag{5b}$$

Finally, $\mathrm{S}_{\mathrm{hol}}$ and S_{ev} are Boolean features that define if the target time slot H is, respectively, during the holidays or specific events periods. For the latter, Table II identifies the relevant list of events for the region.

3) Model definition and hypothesis space: This section presents different types of models trained in the frame of this article with a brief description of each ML technique and the justification of the choices.

The first approach uses Linear Regression (LR) model. This method is a reference and is often used as a first baseline. The model assumes that the mapping between x and y is linear. This method allows to avoid over-fitting, as the prediction error variance is limited by the number of parameters to fit. However, the consequence can be an significant bias.

The second method is the Support Vector Machine Regressor (SVR) [4, Ch. 5]. When addressing binary classification problems, SVM outputs a decision boundary by finding an optimal separating hyperplane that maximizes the distance between the hyperplane and the closest observations on either side. The main added value compared to LR is that it can also manage Non-linearly separable data, mapping them to a high dimensional feature space where a linear decision surface can be applied thanks to the kernel trick. Over-fitting is managed by controlling the margin size between the hyperplane and the closest observations.

The third method is the RF Regressor (RFR) [4, Ch. 6-7]. RF is a meta estimator (ensemble method) that fits a number of classifying decision trees (DT) on various (bootstrapped) sub-samples of the dataset and uses averaging to improve the prediction accuracy. This approach allows to limit over-fitting, by building a strong learner from many weak learners (DT).

4) Evaluation metrics: The trained model is evaluated using an appropriate evaluation metrics. Accounting for the type of ML problem addressed here and in cooperation with operational staff, Mean Absolute Error function (MAE) was selected as the loss function to evaluate the prediction error, i.e., $L(f(x), y) = |f(x) - y|$. Consequently, the performance metric was defined as

$$MAE = \frac{1}{N}\sum_{i=1}^{N} |f(x_i) - y_i|. \tag{6}$$

C. Model training

1) Dataset split for training and testing: Each model selected in section III-B3 was trained using the features presented in Table III. The dataset D was split into a training and a test set: 2019 and 2021 SIV1 data were used for model training and hyper-parameters tuning (D_{train}), while 2022 SIV1 data was used for testing (D_{test}). This temporal split was done to avoid data leakage from future to past and to allow evaluation of the models on unseen data. As already stated, only data between 5AM and 8PM TU have been considered (other slots were dropped). The performance of the ML models were compared against a baseline prediction that is currently used by Nice TCC for operations presented in section III-C3.

2) Model hyper-parameters tuning: A ML model has several settings, the hyper-parameters, that control the behaviour of the algorithm. Their optimization increases the prediction results of the model and hence have to be properly tuned. The training set cannot be used to both training and hyper-parameters tuning because it could yield over-fitting (fitting exactly the training set). Therefore, a dedicated validation set (different from the test set) is needed for hyper-parameters tuning. To avoid splitting the training set again and to take maximum advantage of available data, a 5-fold cross validation splitting strategy has been implemented: it splits the training set into five different subsets. The model is then trained five times in the following way: on training trial $i \in [1, \ldots, 5]$, the model is evaluated on the i-th subset, remaining data is used for training. The global performance of the model is defined as the mean of the performance results of each trial.

Many methods are available for hyper-parameters optimization [2, Ch. 11]: a basic one is to define a grid of hyper-parameters values and to evaluate every possible setting. The main drawback of this systematic method is that its extremely time consuming. An alternative is a random search that consists of evaluating randomly a defined number of possible settings on the grid. Of course, this method is less time consuming than the previous one but can clearly miss the best settings. Therefore, Bayesian optimisation method has been selected, being a good compromise between the alternatives. Its purpose is to select the next settings to evaluate based on to the results obtained from the previously evaluated settings.

In this study, Optuna framework is used for each model hyper-parameters tuning with a Bayesian optimisation method

TABLE IV
OPTIMAL HYPER-PARAMETERS.

Model	Hyper-parameter	Value
LR with Lasso	λ	0.0001249
SVR	[kernel, C]	['rbf', 7620.59825]
RF Regressor	[n_estimators, max_depth, $max\_samples$, $max\_features$, $bootstrap$]	[1093, 10, 0.68, 'auto', True]

[25]. Optuna offers the capacity to handle different sampling methods to define the best settings to evaluate: for instance, Tree Parzen Error (TPE) that samples each hyper-parameter independently [26], or Covariance matrix adaptation evolution strategy (CMA-ES) that exploits the correlations among the parameters [27]. Optuna can even handle a mix of both options. In our case, default setting TPE sampler is selected.

3) Nice TCC baseline: Nice TCC baseline predicts the entry counts (EC) of a given slot H by averaging the data available for the same time slot of the same day of the week, the same week number in the previous years. If EC is considered a function of H, DoW, WN, and year (YN), denoted in short by EC(YN), and considering available data excluding 2020 due to COVID-19, the entry count for a time slot in 2022 is predicted as

$$\mathrm{EC}(2022) = \frac{1}{2} * [\mathrm{EC}(2021) + \mathrm{EC}(2019)] \,. \tag{7}$$

IV. RESULTS

A. Optimal hyper-parameters

As discussed in section III-C2, a Bayesian optimisation method is implemented for each model hyper-parameters tuning. Table IV presents the hyper-parameters selected for tuning and their values obtained by the optimization process.

LR model regularization strength is controlled by the selected regularization technique (Lasso, Ridge, Elasticnet) and the penalty factor called λ. SVR model has its own regularization parameter C that is inversely proportional to the regularization strength of the model. SVR model kernel hyper-parameter also allows to select between many possible kernels to map the data into another dimensional feature space (Gaussian , polynomial, linear, sigmoid and more.).

RFR model utilizes many hyper-parameters:

- $n\_estimators$ controls the numbers of DT trained to build the RF model;
- max_depth controls the maximum depth of each tree;
- $max\_samples$ defines the number of samples to draw from D to train each estimator;
- $max\_features$ defines the number of features to consider when looking for the best split;
- $bootstrap$ to activate bootstrapping of samples.

B. Performance of the models

1) Evaluation on global metrics: Table V summarizes the performance of the various models evaluated on the test dataset by checking the MAE and standard deviation (σ) metrics. LR with Lasso regularization was compared to LR with Ridge and without regularization and it allowed to obtain slight performances improvements. However, results show that it performs poorly compared to Nice TCC baseline. This could be due to LR function space being too restrictive, resulting in the model under-fitting the training set. SVR results are more interesting introducing a global performance improvement of around 14% compared to the Nice baseline.

The RFR model demonstrates superior performances compared to other models (more than 40% MAE improvement compared to LR and more than 10% MAE improvement compared to SVR on the training and test sets) and introduces a global performance improvement of around 25% compared to the Nice baseline. In addition, a 95% confidence interval ($CI_{95\%}$) of error on test set is also computed. It points out that RFR $CI_{95\%}$ on the test set is also 25% narrower than Nice baseline. As the RFR model shows the most promising results, next sections will be dedicated to a closer analysis of this model.

2) Evaluation on illustrative examples: Figure 5 illustrates the RFR entry counts prediction 4 hours ahead on a test set specific day and compares its performance with actual counts and Nice TCC baseline. The figure shows the RFR model overall improvements for each slot: out of the fifteen predicted slots of the day, only one (1PM) seems to perform significantly less accurately than the Nice TCC baseline.

In fact, RFR model is able to capture from our data the general trend of the daily traffic specifically with a general double Gaussian distribution with 2 modes around 9AM and 2-3PM. From an operational perspective, this behaviour seems to correlate well with two usual peaks of traffic, as well as the usual drops in traffic early in the morning, late in the afternoon (linked to the sunrise and sunset) and during the lunchtime break (for this day around 11AM-12PM slots). The model, by nature, limits the variance so the noise around this double Gaussian will be minimal. However, considering this example,

TABLE V
PERFORMANCE METRICS.

Model	Train set MAE	Test set MAE	σ	$CI_{95\%}$
Nice TCC baseline	NA	3.23	4.75	[3.08, 3.38]
RF regressor	2.069	2.446	3.579	[2.332, 2.558]
Linear Regressor	4.17	4.24	5.35	[4.05, 4.34]
SVM regressor	2.42	2.77	3.92	[2.65, 2.90]

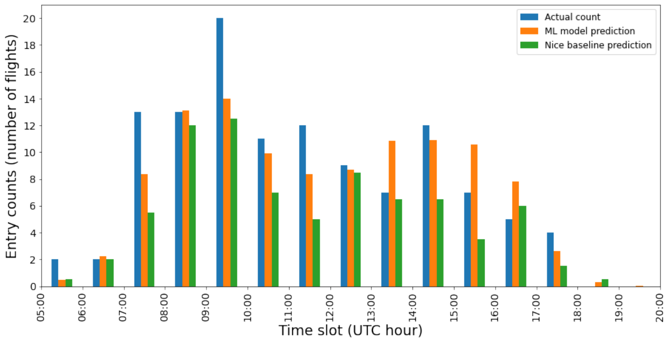

Fig. 5. Illustrative example of results comparison for Thursday, 12th of May 2022.

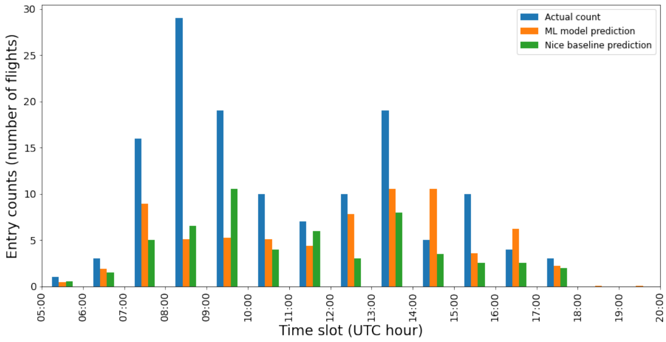

Fig. 6. Illustrative example of results comparison for Thursday, 30th of April 2022 - Cuers rallye.

the actual counts are not as smooth, presenting some noise at 7AM and 11AM.

Another illustrative example is presented in Fig. 6 showing the RFR entry counts prediction of 2022, 30th of April and comparing its performance with actual counts and the Nice TCC baseline. The figure shows very poor performances of both models. After analysis with operational staff, it was noted that Cuers Air Rallye was taking place on that day and a leg of the rallye was crossing SIV1. Neither models were in the position to identify this specific event. Indeed, thanks to operational support, an initial draft of specific events impacting GA traffic was designed in Table II but it now seems obvious that the work is incomplete and needs further improvements.

C. Analysis of the results

A main drawback of the ML algorithms is their lack of explainability. This section intends to better explain ML algorithm predictions concentrating on the RFR model.

A way to better understand a ML model is to focus on the impact of the various features on the prediction. A useful tool related to RF algorithms is the feature importance metric that computes the relative importance of each feature. The importance of a feature is basically the average of the contribution of the feature to the quality criterion reduction (MAE and variance reduction here) on each node of each DT of the RF. Formally, it is computed as the (normalized) total reduction of the criterion brought by that feature. The drawback of this method is the tendency to select numerical and categorical features with high cardinality. The result is shown on Fig. 7.

It confirms that H is the most important feature (more than 50%). This is consistent with preliminary analysis and confirms the link between pilot intent information and entry counts in Nice SIVs (e.g. lunch break, sunrise and sunset impacts, etc.). The second group of features has a lower relative impact on the prediction: meteorology features, WN features, and DoW importance range from 1% to 12%. Meteorology features impact can be explained by the fact that Cannes and Nice meteorology stations are not the most adequate for this SIV. Finally, the last group is related to holidays and specific events that seems to have a very low impact on our model, which is surprising and not aligned with the operational experience.

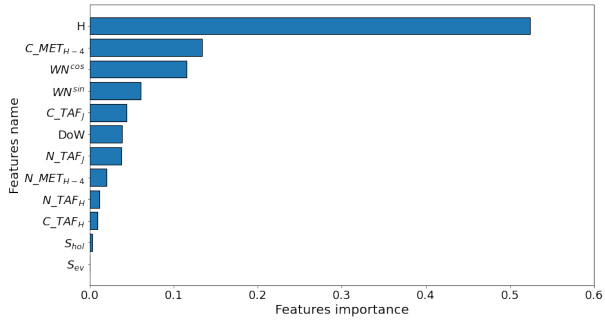

Fig. 7. RFR model features importance.

This last result must be investigated carefully as some features can have an indirect impact not accounted for in this metric : S_{hol} and S_{ev} are Boolean and can be penalized by their low cardinality. To consolidate it, permutation feature importance method has been implemented. This method measures the predictive value of a feature for any estimator by evaluating how the prediction error increases when a feature is not available. This unavailability is implemented, for each feature, by shuffling randomly the values of the feature in the original dataset (D_{test}). The predictions with the original model on the "shuffled" dataset are compared through MAE metric indicating the importance of the feature. The above analysis confirms the findings shown in Fig. 7, including the poor impact of S_{hol} and S_{ev} features. This result can probably be attributed to the limited set of data available (two years and very few days per year related to these events), which is not enough to identify patterns.

V. Conclusions

The objective of this study was to present a methodology to predict GA traffic four hours in advance for ATFCM purpose based on a ML approach. The methodology consists of exploring surveillance, meteorological and calendar data and to train many types of ML models to evaluate their performance on Nice SIV1.

Results using a RF model trained on two years of historical traffic and weather data showed a reduction of the entry counts prediction error around 25% on SIV1 in comparison with current Nice TCC baseline. This is a promising result, obtained using only limited data for this extremely challenging sector characterized by lean meteorological prediction information. It clearly validates the proposed ML approach.

In future work, we intend to address the problem with move advanced Deep Learning techniques: RNN should allow to explore temporal dynamic behaviour of our data. Moreover, a Bayesian Neural Network [28] could be implemented on top to quantify the uncertainty of our ML prediction, which would highly help operations for decision. Other areas to be explored include: a global validation approach on other Nice SIV or on other DSNA TCC, new data exploration (e.g. wider dataset, detailed COVID-19 period analysis, further analysis on holidays and specific events), meteorology scale refinement to challenge the experts' keywords evaluation which could be biased, assessment of new features (e.g. flying schools booking slots, flights trajectory, commercial air traffic), models explainability to increase operational staff confidence (e.g. using a method based on Shapley values).

Acknowledgment

This work was carried out in co-operation with DSNA. The authors thank Nice TCC staff for providing operational expertise and data.

References

[1] E. Fernàndez, J. Cordero, G. Vouros, N. Pelekis, T. Kravaris, H. Georgiou, and S. Ayhan, "Dart: A machine-learning approach to trajectory prediction and demand-capacity balancing," in *Seventh SESAR Innovation Days*, 2017.

[2] I. J. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

[3] D. Jurafsky and J. Martin. (2021) Speech and language processing (3rd ed. draft) stanford course - appendix a.

[4] A. Geron, *Hands-on machine learning with Scikit-Learn*. Sebastopol, CA: O'Reilly Media, 2019.

[5] Z. Wang, A. Liang, and D. Delahaye, "A hybrid machine learning model for short-term estimated time of arrival prediction in terminal manoeuvring area," *Transportation Research Part C Emerging Technologies*, vol. 95 (2018), pp. 280–294, 07 2018.

[6] W. Zeng, X. Chu, Z. Xu, Y. Liu, and Q. Zhibin, "Aircraft 4d trajectory prediction in civil aviation: A review," *Aerospace*, vol. 9, p. 91, 02 2022.

[7] R. Alligier, D. Gianazza, and N. Durand, "Predicting aircraft descent length with machine learning," in *7th International Conference on Research in Air Transportation*, 06 2016.

[8] R. Dalmau, F. Ballerini, H. Naessens, S. Belkoura, and S. Wangnick, "An explainable machine learning approach to improve take-off time predictions," *Journal of Air Transport Management*, vol. 95, p. 102090, 2021.

[9] M. Heffar, R. Dalmau-Codina, and E. Allard, "Prediction of flight departure and arrival routes with gradient boosted decision trees," in *11th SESAR Innovation Days*, 12 2021.

[10] B. Wang, Z. Zhai, R. Xiong, and B. Gao, "Flight trajectory prediction of general aviation aircraft based on lstm model," in *2021 IEEE 4th International Conference on Information Systems and Computer Aided Education (ICISCAE)*, 2021, pp. 176–180.

[11] C. Le Tallec, D. Taurino, C. Lancia, and J. Verstraeten, "Predicting the future location of a general aviation aircraft," NLR Air Transport Safety Institute and Onera and Deep Blue, Nederlands, Tech. Rep., 05 2014.

[12] J. Bedouet and T. Dubot, "Tactical prediction of the number of control positions with softmax regression and tree search," in *Eighth SESAR Innovation Days*, 12 2018.

[13] I. R. Brito, M. C. R. Murca, M. d. Oliveira, and A. V. Oliveira, *A Machine Learning-based Predictive Model of Airspace Sector Occupancy*. AIAA, 2021.

[14] A. Jardines, M. Soler, and J. García-Heras Carretero, "Data-driven occupancy prediction in adverse weather conditions using thunderstorm and traffic observations," in *9th SESAR Innovation Days*, 11 2019.

[15] L. Basora, V. Courchelle, J. Bedouet, and T. Dubot, "Occupancy peak estimation from sector geometry and traffic flow data," in *Eighth SESAR Innovation Days*, 12 2018.

[16] R. Sanaei, B. Pinto, and V. Gollnick, "Toward atm resiliency: A deep cnn to predict number of delayed flights and atfm delay," *Aerospace*, vol. 8, 01 2021.

[17] J. Boucquey, A. Hately, R. Irvine, S. Steurs, F. Gonze, E. Huens, and R. Jungrs, "Probabilistic traffic models for occupancy counting," in *Seventh SESAR Innovation Days*, 2017.

[18] A. Ghobrial, "A model to forecast aircraft operations at general aviation airports," *Journal of Advanced Transportation*, vol. 31, pp. 311–323, 1997.

[19] T. Li and A. Trani, "A model to forecast airport-level general aviation demand," *Journal of Air Transport Management*, vol. 40, p. 192–206, 2014.

[20] Li, Tao and Trani, Antonio, "A least-square model to estimate historical percentages of itinerant general aviation operations by aircraft types and flight rules at an airport," *Journal of Advanced Transportation*, vol. 2017, pp. 1–16, 07 2017.

[21] T. Zakaria, "Estimation of aircraft operations at nontowered airports in the delaware valley region," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1158, pp. 37–46, 1988.

[22] IOWA State University. Iowa environmental mesonet. [Online]. Available: https://mesonet.agron.iastate.edu/

[23] G. Ballester. Ogimet. [Online]. Available: https://www.ogimet.com

[24] D. Chakraborty and H. Elzarka, "Advanced machine learning techniques for building performance simulation: a comparative analysis," *Journal of Building Performance Simulation*, vol. 12, pp. 1–15, 07 2018.

[25] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A next-generation hyperparameter optimization framework," in *Proceedings of the 25rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2019.

[26] J. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl, "Algorithms for hyper-parameter optimization," in *Advances in Neural Information Processing Systems*, J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K. Weinberger, Eds., vol. 24. Curran Associates, Inc., 2011.

[27] N. Hansen and A. Ostermeier, "Completely derandomized self-adaptation in evolution strategies," *Evolutionary Computation*, vol. 9, pp. 159–195, 06 2001.

[28] L. V. Jospin, W. L. Buntine, F. Boussa, H. Laga, and M. Bennamoun, "Hands-on bayesian neural networks - a tutorial for deep learning users," *CoRR*, vol. abs/2007.06823, 2020.