Fourteenth USA/Europe Air Traffic Management Research and Development Seminar (ATM2021)

# Having a Bad Day? Predicting High Delay Days in the National Airspace System

Lu Dai, Mark Hansen
Institute of Transportation Studies
University of California, Berkeley
Berkeley, CA, USA
dailu@berkeley.edu,
mhansen@ce.berkeley.edu

Michael O. Ball
Robert H. Smith School of Business
University of Maryland, College Park
College Park, MD, USA
mball@umd.edu

David J. Lovell
Department of Civil and
Environmental Engineering &
Institute for Systems Research
University of Maryland, College Park
College Park, MD, USA
lovell@umd.edu

***Abstract*—Experiencing high delays is a "bad day" for the National Airspace System (NAS). We apply machine learning algorithms to model the system delay and predict high delay days in the NAS for the 2010s. A broader scope of factors that may affect the system delay is examined, including queueing delays, terminal conditions, convective weather, wind, traffic volume, and special events. We train models to relate the system delay to these features spatially and temporally, and compare the performance of penalized regressions, kernelized support vector regressions, and ensemble regressions. The learned weights of the selected model reveal the spatial pattern and time consistency of the feature importance. Queuing delays, convective weather, and wind are found to be the most significant causative factors for system delays. We then identify high delay days using the model-predicted delay and observe an increasing trend over the past decade. The counterfactual analysis results suggest worsening convective weather after 2014, and a surge in demand in 2013 that was subsequently compensated by increased capacity.**

***Keywords-flight delay prediction; queuing delay; convective weather; machine learning; feature importance***

## I. INTRODUCTION

The U.S. National Airspace System (NAS) is one of the most complex aviation systems in the world, with multiple interrelated components including airlines, airports, system operators, and navigation facilities. Such complexity creates difficulties in management and control. One of the most challenging problems is flight delay. The emerging demand for air traffic and limited airport expansion possibilities have resulted in deteriorating flight on-time performance and increasing delays in the NAS. According to an FAA report [1], flight delays rose by 15% from 2018 to 2019, which was the third straight year at a record high. Growing delays place a significant strain on the NAS and the U.S. economy [2]. From 2012 to 2019, the total cost of delays rose from $19.2 billion to $33 billion, including $18.1 billion in costs to passengers, $8.3 billion in costs to airlines, $2.4 billion from lost demand, and $4.2 billion in indirect cost to other business sectors [1]. We recognize that there was a substantial downturn on flight operations and delay during 2020 due to the pandemic. However, there is a widespread belief that in a relatively short time period, e.g., one to three years, operations levels and delays should return to their prior growth trajectory bringing back delay mitigation challenges.

Experiencing high delays is obviously a "bad day" for the NAS. The need to better understand, quantify, and improve operations of the NAS has been of immediate concern to the FAA's Air Traffic Organization (ATO), and has attracted increasing research attention in recent years. Depending on the objective of the research, various approaches have been developed to predict delay duration, probability of delay, level of delay for a specific flight, airline, airport, or an ensemble of them. Most research focuses on delay prediction for individual flights. Belcastro et al. [3] predicted the arrival delay of individual flights using features derived from aircraft information and weather conditions. With a delay threshold of 15 minutes, the model achieves an accuracy of 74% and a recall of 72%. A more complicated deep learning approach for predicting individual flight delays was proposed by Kim et al. [4]. They first modeled the day-to-day flight delay sequences of a given airport using different recurrent neural network (RNN) architectures. They then fed the predicted delay status of a single day into an individual flight delay model to obtain the final prediction results. At the airport or regional level, Tu et al. [5] developed a probabilistic model to predict the departure delay distributions at Denver airport. Kim and Hansen [6] studied the effects of capacity and demand on flight delays in the New York metropolitan area with simulations through queuing models. Some researchers tackled the flight delay prediction problem at a more aggregated level. Rebollo and Balakrishnan [7] applied a network-based random forest model to predict departure delays with a forecast horizon of 2, 4, 6, and 24 hours, using temporal and spatial delay states of the aviation network as features. For a 2-hour forecast horizon, the average median test error is 21 minutes over the 100 most delayed airport pairs. The test errors grow as the forecast horizon increased. Hsiao and Hansen [8] assumed a simplified NAS network where the 32 busiest commercial airports in the U.S. are considered. They applied econometric models to estimate the system-wide daily average arrival delay from 2000 to 2004 using time-of-day arrival queuing delay, terminal weather, volume, and convective weather as the main explanatory variables. The model not only offered insights into the major causal factors of delay, but also demonstrated the delay propagation effect. They investigated the time-of-day arrival queuing delay, but it was a result of the joint effects induced by 32 airports. The spatial impacts of convective weather on system delay were captured in each $10° \times 10°$ latitude-longitude square region but lacked the time dimension.

While previous studies have yielded valuable insights about causal factors of flight delay and established baselines for predictive performance, none has developed a comprehensive assessment of the system delay considering a wide range of location-specific, time-varying features. Inspired by Hsiao and

This work is sponsored by the FAA through the NEXTOR III Consortium.

Hansen's work [8], this study aims to employ machine learning techniques to model the system-wide daily average delay with a comprehensive feature space extended in the dimensions of time, space, and variety. Our contribution is three-fold: (a) This is the first work to apply machine learning algorithms in modeling and understanding the delay patterns of the NAS for the 2010s, using larger and broader datasets. (b) This study benefits from examining a broader scope of factors that may potentially influence the system delay. Among them, our analysis is unique in its attention to both the effects of realized convective weather and forecasted convective weather, both the effects of surface winds and winds aloft. While current literature only considers aggregated surface wind speed and ignores the winds aloft impact on the en route flight time, our study utilizes the airport runway configuration profile to compute average headwind/tailwind and crosswind speed for each airport, and calculate the winds aloft speed for each Air Route Traffic Control Centers (ARTCC). Our contribution of adding the forecasted weather variables is also critical since most air traffic management actions are based on forecasted weather. (c) Depending on the data resolution, we refine most features into a set of location-specific time-of-day variables to capture the spatial pattern and time consistency of the feature importance. Additionally, the spatial importance is evaluated at either the airport level or the ARTCC level, which has more practical meaning than the regions defined by the latitude-longitude grid.

Our model, which learns to relate system average delay to a wide variety of spatial-temporal features, is of great use for studying flight delays. First, the prediction results given by the model could be used to identify high delay days in the NAS, and examine the trends of predicted high-delay days over time. Second, the learned weights of the model, when finely tuned and fit on the 10-year historical data, are capable of providing the spatial-temporal importance of the features and quantifying their relative importance in affecting the system delay. Flight operators would benefit from having greater foreknowledge of existing delay patterns and how different factors affect the NAS performance. Third, the generalizability of the model, which makes the estimates transferable to a counterfactual context, enhances our understanding of how the system and its environment have changed and affected the system delay over the past decade.

Moreover, the delay predictions are not based on any features related to Traffic Flow Management (TFM), which lays the groundwork for learning the effectiveness of TFM and sheds light on the NAS improvements. Specifically, our predicted high delay days are identified by the system environments only (demand, capacity, weather, etc.), independent of TFM actions taken. The observed high delay days in the real world are the results of system environments and the TFM actions. Comparative study on these two kinds of high delay days should provide meaningful insights into the operational needs of TFM in the NAS. Besides, with forecasted features, the model can offer a reasonable delay forecast for a given day in the future. Flight operators and FAA specialists would value such predictability to plan their responses further in advance, with knowledge of what traffic management actions were taken on a similar historic day and how well they worked [9].

The rest of this paper is organized as follows. In Section II, we introduce the datasets and describe the features by category. Section III discusses the predictive models and the experimental steps. We show the model performance, feature interpretation, prediction results, and counterfactual analysis in Section IV. Finally, conclusions and future work are presented in Section V.

## II. Data and Feature Engineering

In this section, we first describe the datasets used in this study, then introduce features by category. We limit the scope of this study to the Core 30 airports [10] except for the Honolulu International Airport (HNL). To accommodate different time zones, we unified the time zones to UTC-10:00. Hereafter, all the times mentioned in this paper are in the time zone of UTC-10:00. The study period is from 0:00 January 1st, 2010 (UTC-10) to 23:59 December 31st, 2019 (UTC-10). All the datasets and features needed were obtained for this ten-year period.

### *A. Data Sources*

In the FAA Operations and Performance database, the Aviation System Performance Metrics (ASPM) **airport quarter-hour dataset** provides airport level configuration and weather information every quarter hour. There are more than 10 million quarter-hour observations over the selected Core 29 airports from the ten years. The Operational Network **(OPSNET) dataset** provides the daily number of General Aviation (GA) and military operations. We also obtained the ARTCCs boundary shapefile data from the FAA information services.

The **on-time performance dataset** extracted from the Bureau of Transportation Statistics (BTS) TranStats data library, contains the positive arrival delay against schedule (in minutes), canceled flight indicator, diverted flight indicator, and the diverted arrival delay against schedule (in minutes) of each aircraft. By collecting this flight-level data from 2010 to 2019, we obtained 40 million flights that were scheduled to arrive at the selected Core 29 airports. The airport time zone information available from the Master Coordinate table is also retrieved.

The convective weather dataset, the TFM convective forecast (TCF) dataset, and the winds aloft dataset are obtained from different databases maintained by the National Oceanic and Atmospheric Administration (NOAA). The **convective weather dataset** reports the presence of thunderstorms from 2,763 U.S. surface stations. It gives a binary value indicating whether a thunderstorm was observed within 10 miles of a specific surface station at a particular time, but without the echo top height and intensity information. There are 3.5 million data records, and they are updated sporadically. Considering the uniformity and computation cost, we aggregated them into hourly-level data. The **convective forecast dataset** provides graphical representations (polygons) of forecasted convection in the future 4 hours at different echo tops, with a spatial resolution of $0.1 \times 0.1$ degree latitude/longitude. The 4-hour convective forecast is updated every two hours – the forecast of 0:00, 2:00, 4:00, …, 22:00 were issued at 20:00(-1 day), 22:00 (-1 day), 0:00, …, 18:00, respectively. The forecasts cover the 48 contiguous states and adjacent coastal waters in the U.S. There are various forecast models implemented by NOAA over the ten years – collaborative convective forecast product (CCFP, 01/01/2010 – 10/31/2014), Auto CCFP (11/01/2014 – 02/14/2017), and TCF (since 02/15/2017).

The winds aloft dataset records the wind speed (in meters/second) and wind direction (degrees from the north) at different pressure levels ranging from 10 hPa to 1000 hPa at 95 U.S. radiosonde stations. We collected the winds aloft data at the chosen six standard pressure levels – 150, 200, 250, 300, 400, 500 hPa – roughly from FL200 to FL400 (en route altitude). We aggregate the observations to a half-day level, taking noon as the

dividing point. Simply averaging the wind speeds would produce misleading results. Hence, we first decomposed the wind into zonal velocity towards the east and meridional velocity towards the north based on the wind speed and wind direction. We then aggregated them by station, by pressure level, for the two periods of the day. With the zonal velocity and meridional velocity, wind barbs could be drawn at each radiosonde station. Figure 1. shows the wind barbs of the original wind observations in yellow and the aggregated winds in black for each station at 300 hPa on October 25th, 2012, from 0 a.m. to 12 p.m. The dot end is where the wind is blowing to, and the feathers indicate the wind speed. The prevailing winds are blowing from west to east.

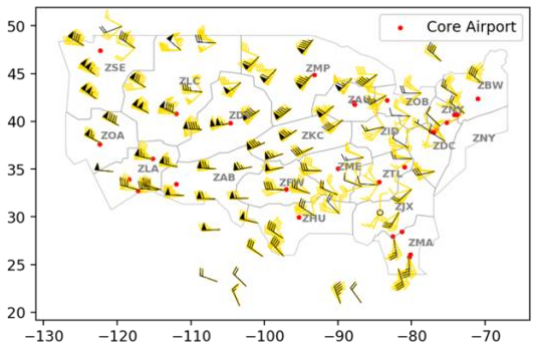


Figure 1. Wind barbs showing original winds (in yellow) and the aggregated winds (in black) at 300 hPa on 10/15/2012, 0 am – 12 pm UTC-10

The **orbital launch logs** are retrieved from the National Aeronautics and Space Administration (NASA) data warehouse. By collecting the space launch reports from 2010 to 2019, we observed 208 launches at the four U.S. launch sites – Cape Canaveral (Florida), Vandenberg Air Force Base (California), Kennedy Space Center (Florida), and Wallops Island (Virginia).

### *B. Delay Metric*

To fulfill the goal of predicting high delay days in the NAS, we first need to define a delay metric for formulating the delay variations of the entire system. Daily average arrival delay (in minutes per flight), which is computed based on average positive delay against schedule for all scheduled arrivals into the Core 29 U.S. airports, is used in this study. Note that:

- This metric accounts for all individual flights scheduled to arrive in the selected 29 airports based on the BTS on-time performance dataset, including completed flights, canceled flights, and diverted flights.
- A flight cancellation is counted as a 120-minute delay against schedule. If a flight was diverted, its diverted arrival delay against schedule is treated as the arrival delay.
- Positive delay against schedule is considered in this metric, where (negative) delays of early arrivals are counted as zero.
- Flights with arrival delays greater than 600 minutes are extreme cases likely caused by, for example, maintenance issues or erroneous data. Such flights are removed from the dataset since they are not in the scope of this study.
- Flights are assigned to days based on their scheduled arrival time.

This delay metric becomes the response variable of our models and is comparable day to day. The high delay days will be picked out when a pre-defined threshold, such as 15 minutes, is exceeded. Figure 2 shows the frequency distribution of the daily average arrival delay over the ten years. The right-skewed shape indicates that a few days had extremely high average delays.

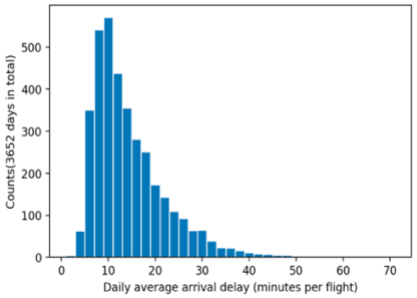


Figure 2. Histogram of the average delay

### *C. Feature Engineering*

To predict high delay days, we construct the feature space at the daily level. According to the literature [8] and theoretical studies, the response variable – daily average arrival delay – is potentially related to queuing delays associated with demand exceeding capacity, terminal conditions, wind, convective weather, traffic volume, and special events like a space launch. These six categories of features are described in the subsections and are summarized in TABLE I. We notice that features at different times of a day or different locations are expected to have different effects on the system performance [8]. Therefore, most variables are broken into multiple location-specific time-of-day variables depending on the data resolution for capturing the spatial and temporal variations.

*1) Queuing Delay:* In a deterministic queuing system, when demand exceeds capacity for a period of time, delay appears. In this study, the cumulative input-output (I/O) queuing diagram is employed to compute the deterministic queuing airport arrival and departure delays based on the ASPM airport quarter-hour dataset.

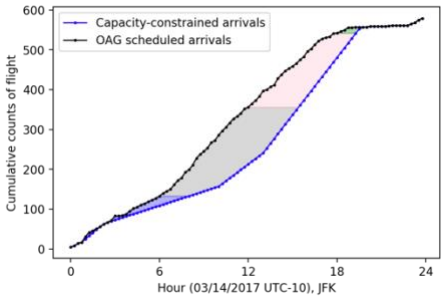


Figure 3. Queuing diagram of JFK airport on 03/14/2017 UTC-10

Figure 3. shows a one-day queuing diagram for JFK arrivals on 03/14/2017. The horizontal axis represents the time at the beginning of each quarter-hour period. As there are 96 quarter hours over a day, the input/output curve comprises 96 points. The black input curve, which represents demand, is expressed by the cumulative OAG scheduled arrivals in flights per quarter hour, including cancellations. The blue output curve, which indicates capacity utilization, is constructed iteratively, by adding the minimum of the Airport Arrival Rate (AAR) and the demand, including unsatisfied demand from previous periods, to the output in the previous period. Therefore, the area between two curves is the total deterministic arrival queuing delay (in flight minutes) for all flights arriving at this given airport on the given day. The queue disappears when the two curves overlap.

The queuing delays at different times of a day are expected to have different effects on the system delay. In addition, the reliability of the queuing delay estimates based on the schedule and called rates varies over time. Thus, we segmented the area between the curves by four time periods of the day: 0:00-6:00, 6:00-12:00, 12:00-18:00, and 18:00-24:00. For example, the gray area represents the total deterministic queuing delay (in flight minutes) for all flights scheduled to arrive at the given airport between 6 a.m. and 12 p.m.

To ensure the continuity of the operations, we generated a ten-year queuing diagram for each of the 29 airports. Both the arrival queuing delays and the departure queuing delays are considered in this study. The average deterministic queuing delay (in minutes per flight) is calculated by dividing the total queuing delay (in flight minutes) by the total number of scheduled flights during a specific period. Specifically, we included the airport-specific time-of-day average arrival/departure queueing delay ($AQD_i^t$ *or* $DQD_i^t$, in minutes per flight) and the daily systematic average arrival/departure queuing delay ($\overline{AQD}$ *or* $\overline{DQD}$, in minutes per flight) in the feature space. They are defined as follows:

$$AQD_i^t \text{ or } DQD_i^t = \frac{d_i^t}{n_i^t} \quad (1)$$

$$\overline{AQD} \text{ or } \overline{DQD} = \frac{\sum_i^{29}\sum_t^4 d_i^t}{\sum_i^{29}\sum_t^4 n_i^t} \quad (2)$$

where $n_{it}$ is the total number of flights scheduled to arrive at/depart from airport $i$ during the time-of-day period $t$, $d_i^t$ is the total deterministic arrival/departure queuing delay for all these $n_i^t$ flights, with units of flight minutes. $AQD_i^t$ and $DQD_i^t$ form the feature vectors $\boldsymbol{AQD}$ and $\boldsymbol{DQD}$, respectively, each with a dimension of $29 \times 4 = 116$. We also incorporate a quadratic term of the daily average arrival/departure queuing delay – $\overline{AQD}^2, \overline{DQD}^2$ to capture the concave relationship between observed delay and queuing delay [8].

*2) Terminal Conditions:* Though the queuing delay features may already capture the airport and weather conditions through AAR/ADR, Hansen and Hsiao [8] found that called rates usually are set higher relative to actual capacity under low capacity conditions such as instrument conditions or windy conditions. Therefore, for a given time-of-day period and airport, we computed the proportion of time an airport is under instrument flight rules (IFR): $\boldsymbol{I}$. To capture the expected nonlinearity of impacts of various visual conditions on delay, the visibility (in statute miles) and ceiling variables (in 100 feet) are discretized into four continuous variables: $\boldsymbol{V_k} = (\boldsymbol{V_1}, \boldsymbol{V_2}, \boldsymbol{V_3}, \boldsymbol{V_4})$ and $\boldsymbol{C_k} = (\boldsymbol{C_1}, \boldsymbol{C_2}, \boldsymbol{C_3}, \boldsymbol{C_4})$ – based on the criteria set for defining low IFR, IFR, marginal VFR, and VFR[11]. Given the airport-specific quarter-hourly arrival runway configuration, we derived two feature vectors $\boldsymbol{RWY}$ and $\boldsymbol{RWY_f}$ to capture the effect of flow pattern change caused by the change of airport runway configuration. $\boldsymbol{RWY}$ is the airport-specific time-of-day vector characterizing the frequency of runway configuration change when the used arrival runways are different between consecutive quarter hours, including opening a new runway or closing an in-use runway. $\boldsymbol{RWY_f}$ characterizes the frequency of the full change of runway configurations if none of the used arrival runways overlap between consecutive quarter hours, such as flipping the airport under windy conditions.

*3) Convective weather:* Convective weather affects the capacity of air traffic resources and thus impacts delay. We derived three types of features to control the convection-related impacts on delay: thunderstorm observation in regional and local levels, convective forecast, and military operations. The convective weather dataset records the presence of thunderstorms at each U.S. surface weather station. Thus, for a given time-of-day period, we calculated the proportion of weather stations in each ARTCC reporting thunderstorms, $\boldsymbol{TSr}$. For a given day and airport, we computed the proportion of scheduled arrivals during the thunderstorm period, $\boldsymbol{TSl}$. The thunderstorm period is defined based on a two-hour expansion of the thunderstorm presence (timestamp) reported by the subject airport weather station. The ARTCC-specific time-of-day feature vector $\boldsymbol{TSr}$ and the airport-specific feature vector $\boldsymbol{TSl}$ are expected to capture the thunderstorm impacts on delay at the regional and local levels, respectively.

The convective weather forecast may also affect the system performance. After preprocessing the convective forecast dataset, we first merge all the forecasted convection polygons into a single cascaded union for a given time-of-day period. We then overlay the ARTCC boundary shapefile and segment the union polygon into multiple areas based on control center boundaries. An example illustrating this segmentation as colored regions is given in Figure 4. Finally, we divide the forecasted area by the total area of each control center and obtain the ARTCC-specific time-of-day area ratio of convection forecasts, $\boldsymbol{TCF}$. In Figure 4, the $TCF$ for the Denver en route center (ZDV in purple) is 1, while the $TCF$ for the Salt Lake City en route center (ZLC in green) is 0.36. Additionally, to control the discrepancy between different convective forecast products over the analyzed period, we create dummy variables – $\mathbb{I}(CCFP)$ and $\mathbb{I}(ACCFP)$ – for different periods in which different products are effective (see Section II.A).

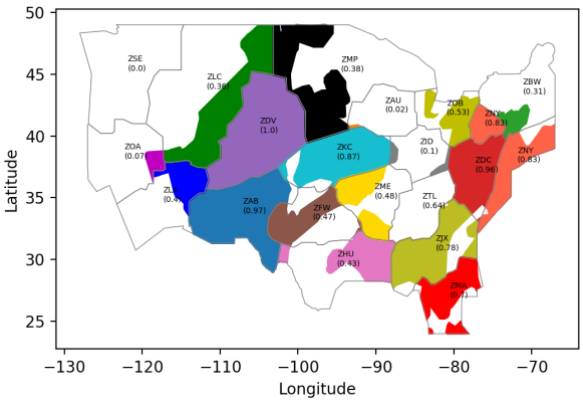


Figure 4. ARTCC-specific convection forecasts

Given that military activities are usually for training purposes and confined to times when weather is favorable, we incorporated the daily number of military operations $M$ as a proxy variable for indicating en route weather conditions. We found a significant difference in the military activities on different days of the week. Thus, a day-of-week dummy variable $\mathbb{I}(\boldsymbol{dow})$ is also added to control for this variation.

*4) Wind:* We considered both the effects of surface winds on terminal landing performance, and the effects of winds aloft on en route cruise performance.

The ASPM airport quarter-hour dataset provides surface wind speed (in knots), wind angle (in degrees), and arrival runway configuration. For each quarter hour, we first apply trigonometric calculations to compute the headwind/tailwind speed and crosswind speed for all the arrival runways listed in the runway configuration profile, and then take the average value. For the variable wind in which the wind angle was not available, we set the headwind/tailwind speed and the crosswind speed as $\sqrt{2}/4 \times$ wind speed. When the wind is a headwind, the tailwind is set to zero, and vice versa. After we obtained the quarter-hourly headwind/tailwind speed and crosswind speed, we aggregate them by the time of day and calculate the average for each airport. Therefore, these airport-specific time-of-day average headwind speed $\boldsymbol{HW}$, average tailwind speed $\boldsymbol{TW}$, and average crosswind speed $\boldsymbol{CW}$ will be used as features to control the effects of surface winds.

After preprocessing the winds aloft data from NOAA (Section II. A), only one aggregated wind was "observed" for each radiosonde station $s$, at each selected pressure level $p$, and for one of the two half-day periods $t$. We denote the wind as a vector of zonal velocity towards the east and meridional velocity towards the north, $\{u_{spt}, v_{spt}\}$. To construct the winds aloft features at the ARTCC level, we further aggregate the zonal velocity and meridional velocity over all the stations within the same control center, and obtain the mean zonal velocity, the sample variance of zonal velocity, mean meridional velocity, and the sample variance of meridional velocity $\{\overline{u}_{jpt}, \sigma^2(u_{jpt}), \overline{v}_{jpt}, \sigma^2(v_{jpt})\}$:

$$\overline{u}_{jpt} = \frac{\sum_s^{S_j} u_{spt}}{S_j}; \quad \sigma^2(u_{jpt}) = \frac{\sum_s^{S_j} (u_{spt} - \overline{u}_{jpt})^2}{S_j - 1} \qquad (3)$$

$$\overline{v}_{jpt} = \frac{\sum_s^{S_j} v_{spt}}{S_j}; \quad \sigma^2(v_{jpt}) = \frac{\sum_s^{S_j} (v_{spt} - \overline{v}_{jpt})^2}{S_j - 1} \qquad (4)$$

where $s$ represents the station within the control center $j$. Hence, there is only one wind vector $[u_{jpt}, v_{jpt}]$ for each control center $j$, at pressure level $p$, and during the time-of-day period $t$. Instead of having one wind barb for each station in Figure 1. , we aggregate winds at different stations within each control center.

One approach would be to stop here and directly use the ARTCC-specific, pressure-level-specific, time-of-day winds aloft features as variables, each with 20 centers $\times$ 6 pressure levels $\times$ 2 time periods $= 240$ dimensions. However, we found that the Pearson correlation coefficient between the winds aloft features at different pressure levels ranges from 0.83 to 0.98, suggesting fairly strong and positive associations. Therefore, we chose to apply Principal Component Analysis (PCA) to decorrelate each of the winds aloft features – $\overline{u}_{jpt}, \sigma^2(u_{jpt})$, $\overline{v}_{jpt}, \sigma^2(v_{jpt})$ – over six pressure levels (dimensions). The first principal component accounts for over 90% of the variance and is selected as the projection direction. Then we transform these pressure-level-variant winds aloft features onto the one-dimensional subspace that is capable of capturing the maximum variance of the data representations. For a given time-of-day period and a control center, our winds aloft features are the PCA mean zonal velocity, PCA variance of zonal velocity, PCA mean meridional velocity, and PCA variance of meridional velocity $\{\overline{u}_{jt}, \sigma^2(u_{jt}), \overline{v}_{jt}, \sigma^2(v_{jt})\}$. The vector form is expressed as $\boldsymbol{UW}, \boldsymbol{\sigma^2(UW)}, \boldsymbol{VW}, \boldsymbol{\sigma^2(VW)}$, each with a dimension of 20 centers $\times$ 2 time periods $= 40$.

*5) Traffic volume:* The airport-specific time-of-day OAG scheduled arrivals ($\boldsymbol{SA}$), and the daily number of GA operations ($GA$) are included. These features, like the queuing variables, capture congestion effects.

*6) Special events:* There are periods when unusual activity takes place and may considerably affect the system performance. We create four site-specific dummy variables to indicate days that had a space launch event, $\mathbb{I}(\boldsymbol{launch})$, for the four launch sites in the U.S. Another dummy variable, $\mathbb{I}(Xmas)$, is added to capture the unusual schedules and flight operations caused by holiday travel. The Christmas dummy variable $\mathbb{I}(Xmas)$ is set to 1 if the day is between December 20 and December 25 inclusive.

With all the variables mentioned above, the feature space is constructed for each observation (day). We notice that features at different times of a day or different locations are expected to have different effects on the delay metric [8]. To capture the spatial and temporal variations, most features (in bold) are refined to either the airport level or the ARTCC level for different periods of the day, depending on the data resolution. For example, the average arriving queuing delay feature vector, $\boldsymbol{AQD}$, is airport-specific and varies at different periods of the day. It thus represents $29\ airports \times 4\ time\ periods = 116$ variables in the feature space. The daily feature vector is summarized in TABLE I. , with dimensions specified for each notation.

## III. PREDICTIVE MODELS

In this section, predictive models and the experimental steps are discussed. The task of predicting high delay days seems like a classification problem. However, it is difficult to obtain a reliable ground truth of labeled high delay days, and it is also hard to validate such a model. Therefore, we define a numeric delay metric (Section II.B) from the BTS on-time performance dataset to reflect the entire system performance. Regression-based machine learning algorithms are employed to learn the relationship between the observed delay metric – daily average arrival delay over the selected 29 U.S. airports – and all the features derived in TABLE I. Once the continuous quantity of the delay metric is predicted, we compare it with a pre-defined threshold, such as 15 minutes, to identify high delay days.

### *A. Machine Learning Models*

We wish to answer the following question: by observing solely the queuing delay, terminal conditions, convective weather, wind, traffic volume, and special events on a given day, how well can a machine learning model learn to predict the average delay of the system? We train and compare three types of machine learning algorithms: linear regression (ordinary least squares (OLS), Ridge, Lasso, Elastic net), kernelized support vector regression (SVR), and ensemble regression (random forest, extreme gradient boosting). Ridge, Lasso, and Elastic net are penalized regression methods, which are linear models but regularize the coefficients toward zero. Regularization addresses concerns about variance-bias tradeoff, multicollinearity, sparse data handling, feature selection, and the interpretability of the output. SVR tries to find the optimal separating hyperplane in the multidimensional feature space within a threshold error value (margin). In this study, Gaussian Radial Basis Function (RBF) kernel and polynomial kernel [12] are applied to capture potential nonlinearities. The random forest (RF) model builds shallow decision trees independently, using a random subset of features, on various subsamples of the dataset. Boosting models sequentially grow decision trees and try to reduce the bias by

learning from previous iterations. We opted for extreme gradient boosting (XGBoost), instead of other boosting models, due to its advantages of being able to add a regularizer in the loss function, subsample features, and fast training.

TABLE I. VARIABLE DESCRIPTIONS

| Category | Notation | Description | Variables (Dim.)[a] |
|---|---|---|---|
| Queuing delay features | $\boldsymbol{AQD}$ | Airport-specific time-of-day average arrival queuing delay (minutes per flight) | 29 × 4 |
| | $\boldsymbol{DQD}$ | Airport-specific time-of-day average departure queuing delay (minutes per flight) | 29 × 4 |
| | $\overline{AQD}$ | Daily systematic average arrival queuing delay (minutes per flight) | 1 |
| | $\overline{AQD}^2$ | Quadratic form of the daily systematic average arrival queuing delay | 1 |
| | $\overline{DQD}$ | Daily systematic average departure queuing delay (minutes per flight) | 1 |
| | $\overline{DQD}^2$ | Quadratic form of the daily systematic average departure queuing delay | 1 |
| Terminal condition features | $\boldsymbol{I}$ | Airport-specific time-of-day proportion of time an airport is under IFR | 29 × 4 |
| | $\boldsymbol{V_k}$ | Airport-specific time-of-day discretized (k = 1, 2, 3, 4) visibility; intervals are [0, 1], (1, 3], (3, 5], (5, 10] (statute miles) | 29 × 4 × 4 |
| | $\boldsymbol{C_k}$ | Airport-specific time-of-day discretized (k = 1, 2, 3, 4) ceiling; intervals are [0, 5], (5, 10], (10, 30], (30, 100] (100 feet) | 29 × 4 × 4 |
| | $\boldsymbol{RWY}$ | Airport-specific time-of-day frequency of the change of runway configuration | 29 × 4 |
| | $\boldsymbol{RWY_f}$ | Airport-specific time-of-day frequency of the full change of runway configuration | 29 × 4 |
| Convective weather features | $\boldsymbol{TSr}$ | ARTCC-specific time-of-day proportion of weather stations reporting thunderstorms | 20 × 4 |
| | $\boldsymbol{TSl}$ | Airport-specific proportion of flights that are scheduled to arrive during the thunderstorm period | 29 |
| | $\boldsymbol{TCF}$ | ARTCC-specific time-of-day area ratio of convection forecasts | 20 × 4 |
| | $\mathbb{I}(CCFP)$ | 1 if the day is between 01/01/2010 – 10/31/2014 inclusive, 0 otherwise | 1 |
| | $\mathbb{I}(ACCFP)$ | 1 if the day is between 11/01/2014 – 02/14/2017 inclusive, 0 otherwise | 1 |
| | $M$ | Daily total military operations | 1 |
| | $\mathbb{I}(\boldsymbol{dow})$ | Dummy variables for day of the week | 6 |
| Wind features | $\boldsymbol{HW}$ | Airport-specific time-of-day average headwind speed (knots) | 29 × 4 |
| | $\boldsymbol{TW}$ | Airport-specific time-of-day average tailwind speed (knots) | 29 × 4 |
| | $\boldsymbol{CW}$ | Airport-specific time-of-day average crosswind speed (knots) | 29 × 4 |
| | $\boldsymbol{UW}$ | ARTCC-specific time-of-day PCA mean zonal velocity (m/s) | 20 × 2[b] |
| | $\boldsymbol{\sigma^2(UW)}$ | ARTCC-specific time-of-day PCA variance of zonal velocity (m/s) | 20 × 2[b] |
| | $\boldsymbol{VW}$ | ARTCC-specific time-of-day PCA mean meridional velocity (m/s) | 20 × 2[b] |
| | $\boldsymbol{\sigma^2(VW)}$ | ARTCC-specific time-of-day PCA variance of meridional velocity (m/s) | 20 × 2[b] |
| Traffic volume features | $\boldsymbol{SA}$ | Airport-specific time-of-day OAG scheduled arrivals | 29 × 4 |
| | $GA$ | Daily total GA operations | 1 |
| Special event features | $\mathbb{I}(\boldsymbol{launch})$ | Launch site-specific dummy variables for days that had a space launch event | 4 |
| | $\mathbb{I}(Xmas)$ | 1 if the day is between 12/01 – 12/25 inclusive, 0 otherwise | 1 |

a. 29 airports; 20 ARTCCs; 4 time periods of the day: 0:00-6:00, 6:00-12:00, 12:00-18:00, 18:00-24:00 (*UTC-10*)

*b.* 2 times periods of the day: 0:00-12:00, 12:00-24:00 (*UTC-10*)

### *B. Experimental Steps*

Some machine learning algorithms are sensitive to the range and distribution of attribute values, and thus may not work well in the presence of outliers. Therefore, it is desirable to remove some outlier observations from the experiment. After all the variables are derived, we applied the 3-interquartile range (IQR) rule for each of the features to detect outliers. We set the scale number at 3 to make the decision range more inclusive and thus more conservative in detecting outliers – any datum that lies beyond 4.7 standard deviations of the mean would be considered an outlier. We removed 42 days that have detected outlier(s) in their feature vectors. In addition, we removed 20 days with daylight savings time transition since operations data are usually unreliable on these days. With outlier days excluded, there are 3,590 days (observations) left in the 10-year analysis period.

After data preprocessing, we randomly split the whole dataset into a training set (80%) and a testing set (20%). Then we standardize all numerical features on the training set and re-use the scaling parameters (mean and standard deviation for each feature) to transform the testing set. Next, we implement five-fold cross-validation on the training set to do an exhaustive search over a specified parameter grid for each candidate model listed in Section III.**Error! Reference source not found.**. The mean absolute error (MAE) is chosen to be the loss function and also the evaluation criteria for model comparison through the experiments. Finally, we fit models with the selected hyperparameters on the entire training set and evaluate the performance on the testing set.

Different algorithms evaluated on the exact same testing set should be comparable since we split the training set and testing set with the same random seed. However, it is hard to tell whether the difference of the model performances on this particular testing set is real or a result of a statistical fluke – if we change the random seed, the model performance on a different testing set may change. Statistical hypothesis tests are designed to address this problem. The null hypothesis is that the two models have equal performance, suggesting that the performance difference between the two models is likely due to a statistical chance. We apply the five replications of two-fold cross-validation ($5 \times 2cv$) with a modified paired Student *t*-test [13] to compare the performance of every two *fine-tuned* models. The *p*-value for each model pair is compared with a pre-defined significance level, $\alpha = 0.05$ in this study. If the *p*-value is smaller than $\alpha$, we reject the null hypothesis and conclude that there is a significant difference in the performance of the two models.

## IV. RESULTS AND DISCUSSIONS

In this section, we first report the performance for different models with the 5x2cv paired *t*-test results. Then we provide interpretations of the learned weights of the selected model. The importance of each group of features is also estimated. Lastly, the prediction results of the selected model are used to identify high delay days. We present the ten-year trend of the high delay days, and conduct the counterfactual analysis to investigate how the system and its environment have changed and affected the system delay over the past decade.

### *A. Model Evaluation and Selection*

Following the experimental steps in Section III.B, all the candidate models are fined-tuned and fit on the whole training set. We then evaluate these eight models, with the optimal hyperparameters, on the exact same testing set and report both the MAE score and root mean squared error (RMSE) in TABLE

II. , with the superior score shaded. Note that for SVR with the polynomial kernel, a degree of 1 is selected by the cross-validated tuning, which is equivalent to a linear kernel.

TABLE II. MODEL PERFORMANCE

| | Linear Regression | | | | SVR | | Ensemble | |
|---|---|---|---|---|---|---|---|---|
| | ***OLS*** | ***Ridge*** | ***Lasso*** | ***Elastic net*** | ***SVR rbf*** | ***SVR linear*** | ***RF*** | ***XGB*** |
| **MAE** | 5.645 | 2.988 | 2.880 | 2.892 | 2.883 | 2.986 | 3.001 | 2.991 |
| **RMSE** | 6.587 | 3.871 | 3.627 | 3.668 | 3.648 | 3.864 | 4.255 | 3.947 |

The MAE across all models ranges from 2.88 to 5.64 minutes, and the RMSE ranges from 3.63 to 6.59 minutes. In general, all models perform considerably well compared to a simple OLS regression. The Lasso regression model generally outperforms other models in terms of MAE score and RMSE score. However, the performance scores do not differ significantly. Therefore, we applied the 5x2cv paired *t*-test to further validate whether such performance difference is statistically significant or not. We calculate the *t*-statistic and the *p*-value for every two models and visualize the *p*-value of each model pair in a masked heatmap in Figure 5. The rows and columns of the heatmap represent the two models that are compared in the hypothesis test. Each cell is colored according to the scale of the *p*-value, in which warmer cells represent larger *p*-values. We assume a significance level of 0.05 for rejecting the null hypothesis that the two models perform equally well on the dataset. The OLS, random forest, and XGBoost have significantly worse performance compared to the other models, as indicated by the dark blue cells in Figure 5. The performance differences between Ridge, Lasso, Elastic net, and kernelized SVR are not statistically significant since their *p*-values are greater than the significance level, and we fail to reject the null hypothesis of equal performance.

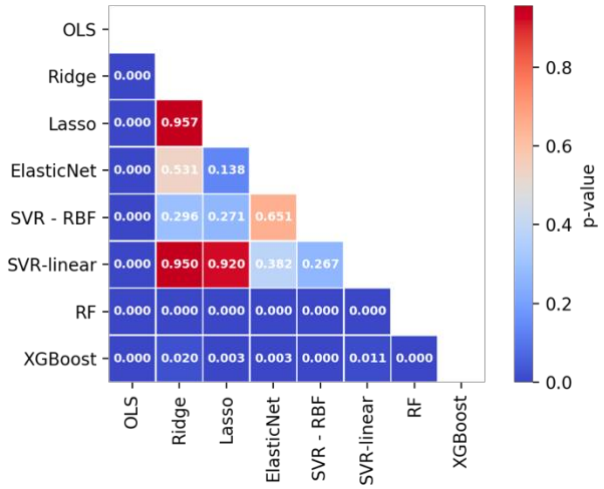


Figure 5. Heatmap showing the p-value of the model performance t-test

In summary, Ridge, Lasso, Elastic net, and kernelized SVR outperform other candidates and have the similar mean performance. There are two main reasons for this: (a) our data has a linear shape in the space. Even if we tune the SVR with a polynomial kernel, a degree of 1 is still preferred by the model to optimize the score; (b) we have a large number of features but relatively less training data – it is usually the case that linear models and SVR outperform tree-based models in such circumstances. These models have an additional advantage: for a regression problem, the range of prediction an RF model can make is bound by the highest and lowest values in the training data. This becomes problematic if our training and testing sets differ in their range. The RF model cannot extrapolate new samples out of the scope of the seen data in the training set, while linear regression would have no problem making accurate predictions for data outside of the training set.

Given that the model performances of Ridge, Lasso, Elastic net, and kernelized SVR are not significantly different, the simpler model – Lasso – is preferred in this study. The shrinkage and feature selection techniques applied in Lasso help reduce variance without a substantial increase of bias, which is especially useful for our case, as we have a large number of features but not that much training data. Therefore, we choose Lasso regression (with the regularization strength of 0.1 for the $l_1$penalty) as our final model for the subsequent analysis and discussion, to get a good fit on our data, to balance bias and variance, and for better interpretability and computational efficiency.

### *B. Model Interpretation*

Lasso regression increases the model interpretability by eliminating irrelevant features that are not associated with the observed arrival delay variable. With the Lasso model fit on our dataset, there are only 332 (out of a total of 2332) variables having a non-zero weight. Most of them have the expected signs. For the queuing delay features, the learned weights of the average arrival queuing delay and its quadratic term, $\overline{AQD}$ (1.73) and $\overline{AQD}^2$ (-0.36), imply a concave relationship between the observed delay metric and arrival queuing delay. This suggests that the system delay diminishes if there is widespread arrival queuing delay in the system. As suggested by Hansen and Hsiao [8], this could be because of a delay masking effect, making the total delay subadditive if the same delay has multiple causes. This does not seem to be the case for the departure queuing delay variables (the quadratic term of departure queuing delay has a zero coefficient).

The convective weather features, especially the observational thunderstorm-related features, significantly affect the system average delay at both regional and local levels. Here we take a closer look at the spatial variability and temporal consistency of the weight vectors for the thunderstorm features $\boldsymbol{TSr}$ ($20 \times 4 = 80$ variables) and $\boldsymbol{TSl}$ (29 variables). At the regional level, the thunderstorm variables $\boldsymbol{TSr}$ of seven ARTCCs are all zeroed out. For the remaining ARTCCs, some thunderstorm time-of-day effects are also eliminated by the model. In Figure 6. , we present the weights of all variables $\boldsymbol{TSr}$ in a map to show its spatial pattern, with warmer color (white, green, yellow, orange towards red) indicating higher temporal consistency of the thunderstorm time-of-day effects. Specifically, the ARTCCs in white color mean all four time-of-day variables $\boldsymbol{TSr}$ have a negligible impact on the system average delay. Thunderstorms observed between 4 p.m. and 10 p.m local time in the Denver control center (ZDV in green) increase system average delay. Los Angeles, Fort Worth, Memphis, Indianapolis, Jacksonville, and Boston centers (in yellow) have two time-of-day variables showing positive effects. Thunderstorms in Chicago, New York, Washington, Atlanta, and Miami centers (in orange) positively, consistently affect the system average delay for three time-of-day periods, and even for the entire day if it is near the Cleveland control center (in red). This likely is because flows through these centers are most susceptible to disruption by convective weather. These centers are also in areas where Airspace Flow Programs (AFPs) are typical [1]. At the local level, we visualize the weights of the 29

$\boldsymbol{TSl}$ variables as cyan circles attached to each of the 29 selected Core airports (black dots). The black dot without the cyan circle represents that the variable $\boldsymbol{TSl}$ of this airport has zero weight. The circle size is proportional to the magnitude of the learned weights, with a larger circle indicating a greater impact of the thunderstorms at a given airport. Observing the spatial pattern and the magnitude of the learned weights, there is a consistency between the local effects of thunderstorms at the airport level and the regional thunderstorm effects at the ARTCC level. The system average delay is consistently sensitive to thunderstorms spreading along the east coast corridors and nearby areas inland for different periods of the day.

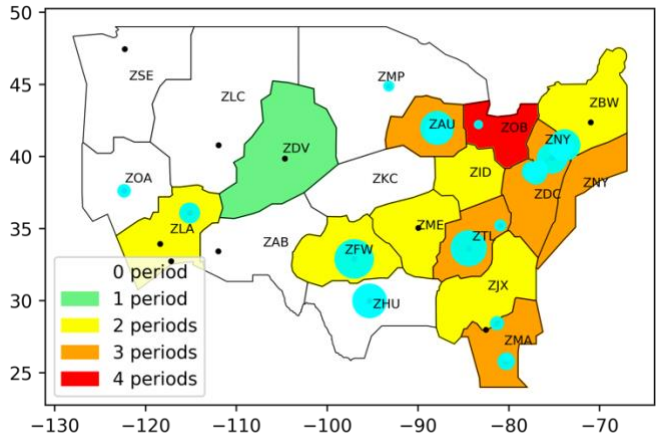


Figure 6. Spatial variability and temporal consistency of thunderstorm features and its importance to the system average arrival delay

## C. Feature Importance

In this study, we focus on the partial dependence that one group of features has on the predicted system delay. Specifically, we answer the question: how can the delay prediction change on average, when a given group of features changes, keeping all other features the same? By measuring the percentage change in prediction $\delta$, we could quantify the importance of a given group of features [14]. The percentage change is defined as:

$$\delta = \frac{E\left[\hat{f}(\boldsymbol{X})\right] - E\left[\hat{f}\left(\boldsymbol{x_c}, \boldsymbol{X_{\setminus c}}\right)\right]}{E\left[\hat{f}(\boldsymbol{X})\right]} \times 100\% \qquad (5)$$

where $\hat{f}(\boldsymbol{X})$ is the predicted average arrival delay provided by the selected model based on the features described in TABLE I. ; $E\left[\hat{f}(\boldsymbol{X})\right]$ is the expectation of delay prediction over all the instances, namely, baseline; $\boldsymbol{x_c}$ represents the group of features for which we want to know their importance to the predicted outcome; $\boldsymbol{X_{\setminus c}}$ are the remaining features, which makes up the total feature space $\boldsymbol{X}$ combined with $\boldsymbol{x_c}$; $E\left[\hat{f}\left(\boldsymbol{x_c}, \boldsymbol{X_{\setminus c}}\right)\right]$ is the expectation of delay prediction over all the instances, with features $\boldsymbol{x_c}$ changed to a specified value.

We first select a group of features $\boldsymbol{x_c}$, such as all the terminal wind features, and replace their feature values with zero knots (no surface wind in the system) for all the observations, while keeping other features unchanged. Next, we applied the trained model to make predictions $\hat{f}\left(\boldsymbol{x_c}, \boldsymbol{X_{\setminus c}}\right)$ for these artificial observations with $\boldsymbol{HW = TW = CW = 0}$. Finally, we calculate the percentage change in prediction, $\delta$, using Equation (5). In this case, we found that the system average delay would have reduced by 20.6% if there were no surface wind in the system, compared to the baseline of 14.23 minutes (the expectation of delay prediction over all the 10-year instances). In other words, this percentage change reflects the importance of the terminal wind to the system delay.

We repeat this procedure for different groups of features and visualize their relative importance in Figure 7. The horizontal axis lists the group of features $\boldsymbol{x_c}$ that we change to reduce the system average delay. The height of the color bar indicates the percentage reduction in predicted delay $\delta$, which is also interpreted as the feature importance. The rank of the feature importance in general matches our expectations. Queuing delay, thunderstorms, and wind have the highest impacts on reducing the system average delay. Eliminating all queuing delays would reduce arrival delay by 27%; arrival queuing delay has a greater delay impact than departure queuing delay. If there are no thunderstorms in the system, either at a regional or local level, the system arrival delay will drop by about 21.5%. Taking out the effects of surface winds on the terminal and the effects of winds aloft en route, would result in a total of 37.6% reduction of the system average delay. As expected, the queuing delay variables may not fully capture the effects of the ceiling, visibility, meteorological conditions, and the change of runway configuration. Ideal visibility and ceiling conditions at the airports would trigger about a 12% reduction in system delay, even while keeping the queuing delay variable constant.

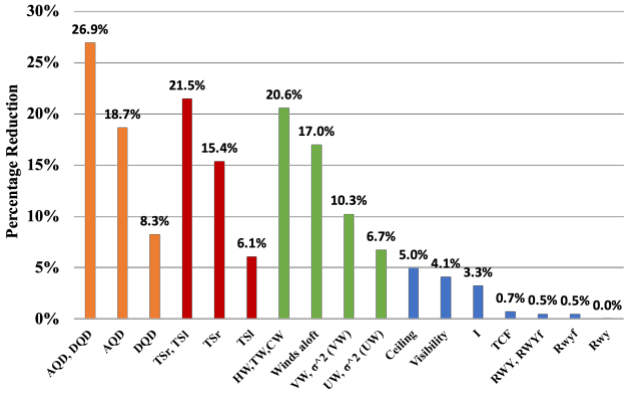


Figure 7. Relative feature importance in reducing system average delay

## D. Predicting High Delay Days

In this section, we identify high delay days using the predicted average arrival delay given by the model. If a day is predicted to have an average arrival delay exceeding a certain threshold, it will be labeled as a high delay day. Note that these high delay days are defined based on the features we describe in TABLE I. such as demand, weather, and are relatively independent of TFM actions taken. Our predicted high delay days may differ from the observed high delay days due to the intentionally ignored TFM effects as well as model errors. We pick two threshold values – 15 minutes and 20 minutes – to assess the sensitivity of the results. Using the 15-minute threshold, 1,327 high delay days are identified over the 10-year analysis period, while only 548 days are labeled as high delay days with the 20-minute threshold.

Moreover, we are more interested in examining the trend of high-delay days, and to what degree the high delay days are increasing over time. Therefore, we calculate the percentage of high delay days for each year and plot it in Figure 8. Different colors represent the two threshold values we used to identify high delay days. Overall, we have an increasing trend of high delay days over the past ten years. The percentage of high delay days has almost doubled since 2010 and has grown rapidly in recent years.

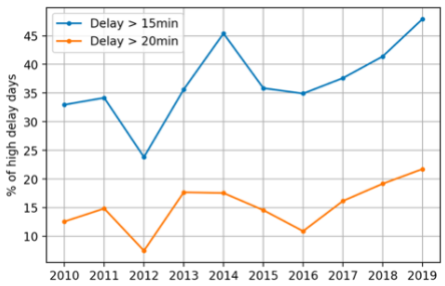


Figure 8. High delay days from 2010 to 2019

### *E. Counterfactual Analysis*

In this section, we investigate how the system and its environment have changed and affected the system delay over the past decade. Our counterfactual analysis quantifies how different factors have contributed to the trend of high delay days shown in Figure 8. Factors of interest include increasing demand, improvement of aviation facilities and infrastructure (capacity), or deteriorating environment (more thunderstorms). The "counterfactual" evaluates what would have happened to the NAS, specifically the trend of high delay days, in the absence of changes in these factors. The impact is estimated by comparing the counterfactual predicted outcome – either average delay or the number of high delay days – to the predicted outcome based on the actual features. For example, if the demand in 2019 had not increased from 2010, keeping all other features the same, how would delay in 2019 have changed? By answering this question, we obtain a counterfactual delay in 2019 (outcome under counterfactual) that can be compared to the 2019 predicted delay based on the actual demand in 2019. The difference between these two outcomes is the impact of the factor – in this example, growing demand in the NAS.

TABLE III. COUNTERFACTUAL ANALYSIS RESULTS

| **Outcome under intervention (2019)** | **Average arrival delay** | | **High delay days (>15min)** | | **High delay days (>20min)** | |
|---|---|---|---|---|---|---|
| | **15.980 min/flight** | | **175** | | **80** | |
| **Features using 2010 data** | ***Outcome*** | ***% change*** | ***Outcome*** | ***% change*** | ***Outcome*** | ***% change*** |
| **Demand** | 14.804 | -7.36% | 146 | -16.57% | 69 | -13.75% |
| **AAR/ADR** | 17.319 | 8.38% | 208 | 18.86% | 96 | 20.00% |
| **Demand & AAR/ADR** | 16.016 | 0.23% | 177 | 1.14% | 81 | 1.25% |
| $\boldsymbol{TSr}$ | 15.084 | -5.61% | 169 | -3.43% | 54 | -32.50% |
| $\boldsymbol{TSl}$ | 15.724 | -1.50% | 176 | 0.57% | 77 | -3.75% |
| $\boldsymbol{TSr}$ & $\boldsymbol{TSl}$ | 14.828 | -7.11% | 155 | -11.43% | 53 | -33.75% |
| $\boldsymbol{TCF}$ | 16.031 | 0.32% | 177 | 1.14% | 84 | 5.00% |
| **Wind**[a] | 16.29 | 1.94% | 188 | 7.43% | 86 | 7.50% |
| $\boldsymbol{RWY}$, $\boldsymbol{RWYf}$ | 16.057 | 0.48% | 175 | 0.00% | 84 | 5.00% |
| $\boldsymbol{I}$ | 15.911 | -0.43% | 175 | 0.00% | 80 | 0.00% |
| $\boldsymbol{C_k}$ | 15.994 | 0.09% | 177 | 1.14% | 82 | 2.50% |
| $\boldsymbol{V_k}$ | 15.987 | 0.04% | 185 | 5.71% | 75 | -6.25% |

a. All the winds aloft features and terminal wind features

Taking the growing demand factor as an example, we first construct the counterfactual scenarios by replacing all the demand data (OAG scheduled arrivals/departures) in 2019 with the demand data in 2010. We then regenerate the queuing delay features with the 2010 demand but keep the 2019 capacities (AAR/ADR). Next, we apply the trained model to the counterfactual feature matrix to predict the daily system average delay in 2019 and identify high delay days with the pre-defined threshold. Finally, we calculate the percentage change in prediction using Equation (5), but with $\boldsymbol{x_c}$ as the counterfactual features. For the prediction year 2019, we repeat this procedure for different groups of the features, using the 2010 features as the counterfactuals. The counterfactual analysis results for this year pair are summarized in TABLE III.

Based on the actual 2019 features, the average arrival delay in 2019 was 15.98 minutes per flight, and 175 days (80 days) are identified as high delay days using the 15-minute (20-minute) threshold. The "outcome" column reports the counterfactual predictions in 2019, with one set of features at a time being replaced with the corresponding 2010 features. The "% change" column shows the percentage change between the predicted counterfactual outcomes for various features and the predictions based on actual 2019 features, according to Equation (5). The first row of the results answers the question in the example: if the demand in 2019 had not increased from 2010, keeping all other features the same, the delay in 2019 would have decreased 7.4%, and the number of high delay days defined by the 15-minute and 20-minute threshold would have dropped 16.6% and 13.8%, respectively. However, if both the demand and the capacity stay the same as they were in 2010, keeping all else equal, the delay in 2019 only increases 0.23%. This suggests that the capacity just kept up with the demand in the NAS over the ten years. TABLE III. also shows that delays would have 7.11% lower in 2019 if convective weather in that year had been the same as in 2010. Counterfactuals involving other features show much smaller delay differences,

The results we present so far are only for one year pair (2010 & 2019). We now extend the counterfactual analysis for every pair of the years from 2010 to 2019. In other words, for every feature of interest (e.g., convective weather), we construct 90 counterfactual scenarios for all the permutations among the ten years. Due to limited space, we only show the all-year-pair counterfactual results for the thunderstorm features ($\boldsymbol{TSr}$, $\boldsymbol{TSl}$) in Figure 9. The horizontal axis represents the predicted year, and the vertical axis represents the year whose data are used to construct counterfactual features. Each cell shows the percentage change in delay, with warmer colors indicating delay increase – the year on the horizontal axis is better; otherwise (blue), the year on the vertical axis is better. The diagonal cells represent the actual situation without counterfactuals, so there is no change in the prediction. Picking one number in the top right corner (-7.1%), it means that if the thunderstorm events in 2010 (year as counterfactual) had been presented in 2019 (predicted year), the system average delay in 2019 would have decreased 7.1%. This result is the same as what we obtained in TABLE III. , and indicates better thunderstorm activities in 2010 for the NAS performance.

The whole heat map shows the results for every year pair. We observe that worsening convective weather has caused increased delays, especially after 2014. This observation helps explain the post-2014 rapidly rising trend in the number of high delay days shown in Figure 8. It is also consistent with observations in the Earth Environmental field that the years from 2015 to 2019 have become the warmest five years ever [15], and extreme weather is getting more frequent, intense, and severe due to climate change [16]. From the all-year-pair counterfactual results for the queuing delay features, we also observe that demand surged in 2013, caused temporary delay increases, but

capacity subsequently caught up. This observation explains the small peak around 2013 – 2014 in the trend of high delay days shown in Figure 8. Besides, delay impacts of year-to-year wind changes have been modest and exhibit no clear trend. Similar to what we obtained for the one year pair in TABLE III. , other features seem to be stable over time with respect to their contributions to flight delay.

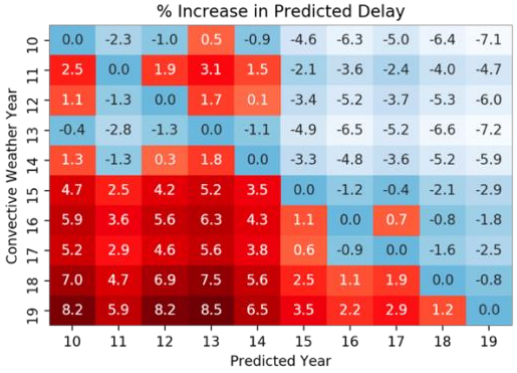


Figure 9. The all-year-pair counterfactual results for thunderstorm features

## V. Conclusions and Future Work

In this study, we apply machine learning techniques to predict flight delays in the NAS using spatial-temporal features, including queuing delays, terminal conditions, convective weather, wind, traffic volume, and special events such as space launches and Christmas holidays. Our system delay metric is based on average positive delay against schedule for all scheduled arrivals into the Core 29 U.S. airports, adjusted to include flight cancellations. It is used to identify high delay days in the NAS by setting a pre-defined threshold, such as 15 minutes. Together with an OLS model as the baseline, seven machine learning models are fine-tuned with five-fold cross-validation and fit on the 10-year historical data. For interpretability, computational efficiency, and a better fit of our data, we choose the Lasso as the final model, with an MAE score of 2.88 minutes.

The learned weights of the model are consistent with conventional wisdom. We found a concave relationship between the system delay and arrival queuing delay, and the spatial importance of the thunderstorm features at both regional and local levels. We further quantify the importance of each group of features by calculating the percentage change in predicted delay after replacing the feature values. The rank of the feature importance in general matches our expectation – queuing delay, thunderstorm, and wind have the greatest impact on the system average delay. We then identify the high delay days in the NAS with two thresholds – 15 minutes and 20 minutes. Overall, we observe an increasing trend of high delay days over the ten-year period, with rapid growth in recent years. To better understand how the system has changed and affected system delay over the past decade, we apply a counterfactual analysis to all the 90 year-pairs. We observe worsening convective weather, especially after 2014, which explains the rapidly increasing trend in the number of bad days from 2015 to 2019. Demand surged in 2013, caused temporary delay increases, but capacity subsequently increased a similar amount. This could explain the small peak appearing around 2013-2014 in the trend of high delay days.

In future work, we will assess the effectiveness of TFM by considering the traffic management actions taken on the predicted high delay days. Our predicted high delay days are identified by the system environments only (demand, capacity, weather, etc.), independent of TFM actions taken. The observed high delay days in the real world can be treated as the outcome under interventions – referred to as TFM actions. Comparative study on these two outcomes should provide meaningful insights into the operational needs of TFM and the improvement of the NAS. This analysis can also be extended to forecasting high delay days in the NAS in the future, using forecasted features instead of realized features. For example, the demand forecasts can be used in our analysis to reconstruct the queuing delay features for delay forecasts. Flight operators and FAA specialists would value such predictability to plan their responses further in advance, with knowledge of what traffic management actions were taken on a similar historic day and how well they worked [9]. It is possible that model performance could be improved by adding more data and relevant features such as precipitation and snow presence.

## Acknowledgment

This work was supported by the FAA through the NEXTOR III Consortium. We are grateful for the input from the project's advisory team, including Brian Bagstad, Kerry Capes, Garry Cohen, Carlos Gonzalez, Leo Prusak, Guillermo Sotelo, and Jim Wetherly. We thank Jonathan Leffler from NOAA for providing us the TCF data archive.

## References

[1] FAA. Air Traffic By The Numbers. 2020: Federal Aviation Administration.
[2] Ball, M., C. Barnhart, M. Dresner, M. Hansen, K. Neels, A. Odoni, E. Peterson, L. Sherry, A. Trani, and B. Zou. Total delay impact study: a comprehensive assessment of the costs and impacts of flight delay in the United States. 2010.
[3] Belcastro, L., F. Marozzo, D. Talia, and P. Trunfio. Using scalable data mining for predicting flight delays. ACM Transactions on Intelligent Systems and Technology (TIST), 2016. **8**(1): p. 1-20.
[4] Kim, Y.J., S. Choi, S. Briceno, and D. Mavris. A deep learning approach to flight delay prediction. In 2016 IEEE/AIAA 35th Digital Avionics Systems Conference (DASC). 2016. IEEE.
[5] Tu, Y., M.O. Ball, and W.S. Jank. Estimating flight departure delay distributions—a statistical approach with long-term trend and short-term pattern. Journal of the American Statistical Association, 2008. **103**(481): p. 112-125.
[6] Kim, A. and M. Hansen. Deconstructing delay: A non-parametric approach to analyzing delay changes in single server queuing systems. Transportation Research Part B: Methodological, 2013. **58**: p. 119-133.
[7] Rebollo, J.J. and H. Balakrishnan. Characterization and prediction of air traffic delays. Transportation research part C: Emerging technologies, 2014. **44**: p. 231-241.
[8] Hsiao, C.-Y. and M. Hansen. Econometric analysis of US airline flight delays with time-of-day effects. Transportation Research Record, 2006. **1951**(1): p. 104-112.
[9] Gorripaty, S., Y. Liu, M. Hansen, and A. Pozdnukhov. Identifying similar days for air traffic management. Journal of Air Transport Management, 2017. **65**: p. 144-155.
[10] FAA. FAA Operations & Performance Data. Federal Aviation Administration.
[11] FAA. Aeronautical information manual. 2011, US Department of Transportation Washington, DC.
[12] Soentpiet, R. Advances in kernel methods: support vector learning. 1999: MIT press.
[13] Dietterich, T.G. Approximate statistical tests for comparing supervised classification learning algorithms. Neural computation, 1998. **10**(7): p. 1895-1923.
[14] Dai, L., Y. Liu, and M. Hansen. Modeling Go-around Occurrence. In Proceedings of the Thirteenth USA/Europe Air Traffic Management Research and Development Seminar (ATM2019), Vienna, Austria. 2019.
[15] Cheng, L., J. Abraham, J. Zhu, K.E. Trenberth, J. Fasullo, T. Boyer, R. Locarnini, B. Zhang, F. Yu, and L. Wan. Record-setting ocean warmth continued in 2019. 2020, Springer.
[16] EPA. Climate Change Indicators: Weather and Climate. 2020 [cited 2021 April 16]; Available from: https://www.epa.gov/climate-indicators/weather-climate.