AIRCRAFT ARRIVAL DELAY PREDICTION USING MACHINE LEARNING METHOD

Flt.Lt. Patara Charnvanichborikarn

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Engineering in Industrial Engineering
Department of Industrial Engineering
Faculty Of Engineering
Chulalongkorn University
Academic Year 2023

การทำนายอากาศยานที่เดินทางถึงล่าช้ากว่ากำหนดด้วยวิธีการเรียนรู้ของเครื่อง

ร.อ.ภัทร ชาญวานิชบริการ

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมอุตสาหการ ภาควิชาวิศวกรรมอุตสาหการ
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2566

Thesis Title	AIRCRAFT ARRIVAL DELAY PREDICTION USING MACHINE LEARNING METHOD
By	Flt.Lt. Patara Charnvanichborikarn
Field of Study	Industrial Engineering
Thesis Advisor	Assistant Professor NANTACHAI KANTANANTHA, Ph.D.

Accepted by the FACULTY OF ENGINEERING, Chulalongkorn University in Partial Fulfillment of the Requirement for the Master of Engineering

..... Dean of the FACULTY OF ENGINEERING
(Professor SUPOT TEACHAVORASINSKUN, D.Eng.)

THESIS COMMITTEE

..... Chairman
(Associate Professor WIPAWEE THARMMAPHORNPHILAS, Ph.D.)

..... Thesis Advisor
(Assistant Professor NANTACHAI KANTANANTHA, Ph.D.)

..... Examiner
(Assistant Professor NARAGAIN PHUMCHUSRI, Ph.D.)

..... External Examiner
(Associate Professor Chansiri Singhtaun, D.Eng)

ภัทร ชาญวานิชบริการ : การทำนายอากาศยานที่เดินทางถึงล่าช้ากว่ากำหนดด้วยวิธีการเรียนรู้ของเครื่อง. (AIRCRAFT ARRIVAL DELAY PREDICTION USING MACHINE LEARNING METHOD) อ.ที่ปรึกษาหลัก : ผศ. ดร.นันทชัย กานตานันทะ

ความล่าช้าของเที่ยวบินนับเป็นความท้าทายที่สำคัญต่อการเติบโตและประสิทธิภาพของอุตสาหกรรมการบินของสหรัฐอเมริกา ซึ่งเป็นจุดสนใจหลักของวิทยานิพนธ์ที่สะท้อนให้เห็นถึงภาคสายการบินทั่วโลก การศึกษานี้มีวัตถุประสงค์เพื่อคาดการณ์ความล่าช้าในการมาถึงของเครื่องบินโดยใช้แบบจำลองการเรียนรู้ของเครื่อง เพื่อจัดการกับความล่าช้าของเครื่องบินซึ่งเป็นความท้าทายที่สำคัญสำหรับอุตสาหกรรมการบินของสหรัฐอเมริกา การวิจัยอาศัยชุดข้อมูลจาก Kaggle โดยนำมาทำการจัดระเบียบข้อมูลใหม่ให้มีความครอบคลุม วิทยานิพนธ์นี้มีการนำเอามุมมองที่สำคัญเกี่ยวกับการรวบรวมข้อมูลสภาพอากาศจากสนามบินปลายทาง ณ เวลาที่เครื่องบินจะมาถึงเข้าไปในแบบจำลอง ดังนั้นการปรับปรุงชุดข้อมูลของวิทยานิพนธ์นี้จะพัฒนาความแม่นยำในการทำนายของเครื่องมากขึ้น วิทยานิพนธ์นี้กำหนดตัวชี้วัดประสิทธิภาพของวิธีการเรียนรู้ของเครื่องทั้งหมดสี่วิธี ได้แก่ Random Forest, CatBoost, Gradient Boosting และ AdaBoost โดยทำการใช้ GridSearchCV ที่นำไฮเปอร์พารามิเตอร์ทุกแบบมาทดสอบอย่างครบถ้วน ผลลัพธ์ที่ได้ตอกย้ำถึงความแม่นยำของแบบจำลอง Random Forest โดยแสดงความแม่นยำที่น่าประทับใจถึง 83% และ F1-score ที่ 0.56 ซึ่งถือเป็นวิธีการที่มีความแม่นยำมากที่สุด ในขณะที่ Gradient Boosting, CatBoost และ AdaBoost มีความแม่นยำ 81.1%, 81% และ 72.6% ตามลำดับ ค่า F1-score สูงสุดที่ 0.56 ซึ่งได้มาจากวิธี Random Forest ตามมาด้วย Gradient Boosting (0.47), CatBoost (0.46) และ AdaBoost (0.45) อย่างไรก็ตาม ข้อจำกัดในการเข้าถึงข้อมูลการจราจรบนเที่ยวบิน สภาพอากาศระหว่างเส้นทางในระดับความสูงของเที่ยวบินที่แตกต่างกัน และรายละเอียดเฉพาะของเครื่องบินนั้นๆ เน้นย้ำถึงความจำเป็นในการปรับปรุงกลยุทธ์การรับข้อมูล เพื่อเพิ่มประสิทธิภาพของการคาดการณ์การมาถึงของเครื่องบินแบบเรียลไทม์ และปรับแต่งแบบจำลองการคาดการณ์เพิ่มเติม เนื่องจากความล่าช้าของเครื่องบินซึ่งส่วนใหญ่ส่งผลมาจากการจราจรทางอากาศและสภาพอากาศนั้น ยังคงเป็นความท้าทายที่สำคัญในปัจจุบัน ดังนั้นแบบจำลองการพยากรณ์เฉพาะทางและการบูรณาการเทคโนโลยีขั้นสูงจึงมีความจำเป็นที่จะนำมาใช้ในการปรับปรุงประสิทธิภาพของการเดินทางทางอากาศ

สาขาวิชา วิศวกรรมอุตสาหการ ลายมือชื่อนิสิต

.....

ปีการศึกษา 2566 ลายมือชื่อ อ.ที่ปรึกษาหลัก

.....

6370221321 : MAJOR INDUSTRIAL ENGINEERING
KEYWORD: AIRCRAFT ARRIVAL DELAY PREDICTION USING MACHINE LEARNING METHOD

Patara Charnvanichborikarn : AIRCRAFT ARRIVAL DELAY PREDICTION USING MACHINE LEARNING METHOD . Advisor: Asst. Prof. NANTACHAI KANTANANTHA, Ph.D.

Field of Study:	Industrial Engineering	Student's Signature .....
Academic Year:	2023	Advisor's Signature .....

ACKNOWLEDGEMENTS

This thesis marks the end of an incredible journey, and I'm immensely grateful to everyone who made it possible. First, I want to express my deepest appreciation to my academic advisor, Assistant Professor Nantachai Kantanantha. Your guidance and unwavering support shaped this research profoundly.

I'm also thankful to my thesis committees for their invaluable insights that enhanced the quality of my work.

To my parents, Sakol and Sripatra Charnvanichborikarn, your encouragement and unwavering support have been my driving force.

To my friends, Pongpisut Kongdan, and Krisda Chugh, thank you for your enduring support. Kamonnat, your belief in finding light in the darkest days has been a constant inspiration.

To everyone who contributed to my growth, big or small, thank you. This thesis is a tribute to your collective support on this remarkable journey.

Patara Charnvanichborikarn

TABLE OF CONTENTS

Page

LIST OF TABLES

Page

LIST OF FIGURES

Page

Chapter 1: Introduction

In this thesis, the main focus is on the arrival delays of the aircraft. The purpose of this chapter is to introduce the broad background of the airline industry, narrowing down to the US airline industry, which is the sample of the data, and highlight key issues as well as the problem statement.

1.1 Global Airline Industry

The airline industry has taken a major setback in 2020 from the COVID-19 pandemic with a decrease of 168 billion USD in revenues. Figure 1 has illustrated the change in the world's passenger traffic from 1945-2022. Among other industries, it was ranked number one with the most substantial loss according to McKinsey & Company. (Bouwer et al., 2022)

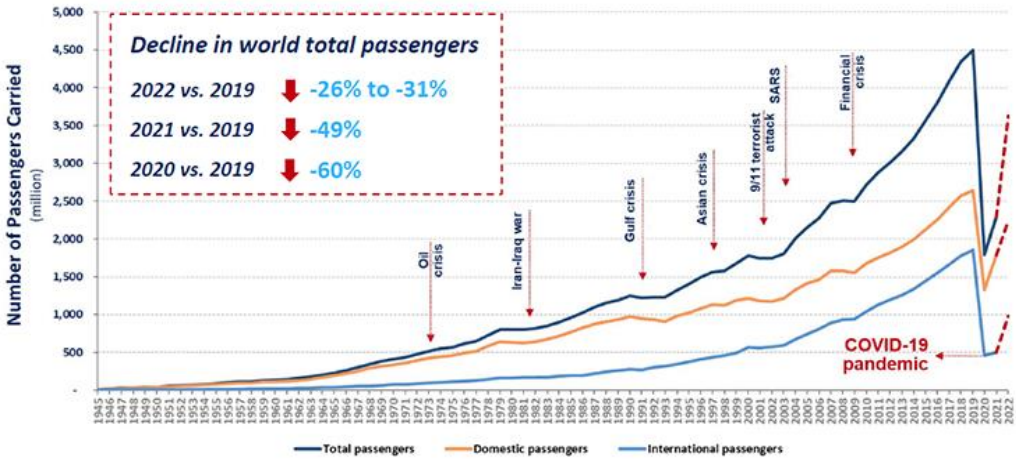


Figure 1: World Passenger Traffic Evolution, 1945-2022 (ICAO, 2021)

As a result of the access to the vaccines, and the lift of regulations worldwide in 2021, there has been a recovery in many businesses, including passenger flights. The global revenues rose by 27% as compared to 2020 but, they were still 44% less than what they were in 2019 (GAO, 2021). Moreover, according to the Federal Aviation Administration (FAA, 2020), the industry was able to fill their aircraft with over 83.4% passenger load factor in 2019 but, in 2021, the number was only 69.2%. On the other hand, cargo airlines, which take a small fraction as a sub-sector of the overall airline businesses, have maintained

positive revenue due to the rise of demand in freight forwards. According to McKinsey & Company, cargo carriers had annual profit averaging 2 billion USD from 2012-2019. Their yields also rose by 40 percent year on year in 2020, and by an additional 15 percent in 2021. Load factors were also significantly increased, by ten percentage points in 2021 as compared to 2019.

In 2022, the Ukraine and Russia conflict raised concerns about global economic stability, particularly in the aviation industry. Flights passing through Russian and Ukrainian airspaces were rerouted, resulting in longer durations and limitations in route planning and scheduling. This led to increased costs due to both a supply shock in energy, higher resource consumption such as increased fuel usage, longer working hours for air crews, and more flight hours for aircraft. As a consequence, these cost increases were reflected in higher plane ticket fares for passengers.

With major recent events impacting the industry, it is obvious that the airline industry is not generating the same revenue. At the same time, it has to absorb the fixed and variable costs, such as aircraft maintenance and employee wages. Therefore, airlines could potentially aim to maximize their operation efficiency and improve cost-effectiveness in response to supply-chain disruptions and a decline in demand.

1.2 US Airline Industry

Considered one of the largest industries in the world, the US airline industry carries 2.5 million passengers per day to and from nearly 80 countries, moves 58,000 tons of cargo per day to and from more than 220 countries, creates jobs for almost 750,000 employees, and powers 28,000 flights across the globe, according to Airline for America. (2022) (America, 2022)

Like any other region in the world, the US was severely affected by the Covid-19 pandemic and airlines are one of the most affected industries. According to Department of Transportation (DOT) statistics, passenger traffic declined 90% in April 2020 as compared to the previous year.

Due to a rise in demand for travels in 2021, the industry had a short rebound and later was slowed down because of the delta variant. Although the cargo airlines have still retained almost the same output and reached the all-time-high in 2021 as seen in Figure 2, the overall airline industry could not secure positive profits after the pandemic as compared to other industries in the US as seen in Figure 3.

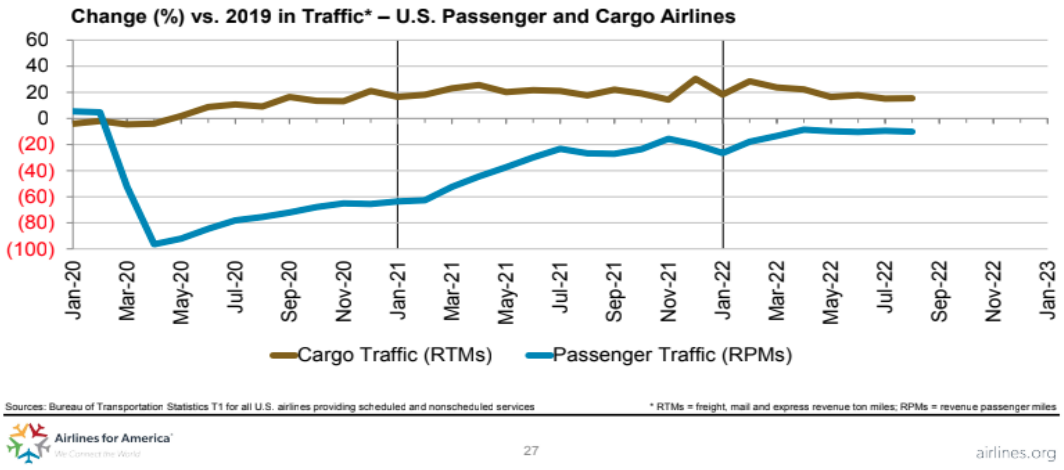


Figure 2: US Cargo Traffic and Passenger Traffic (America, 2022)

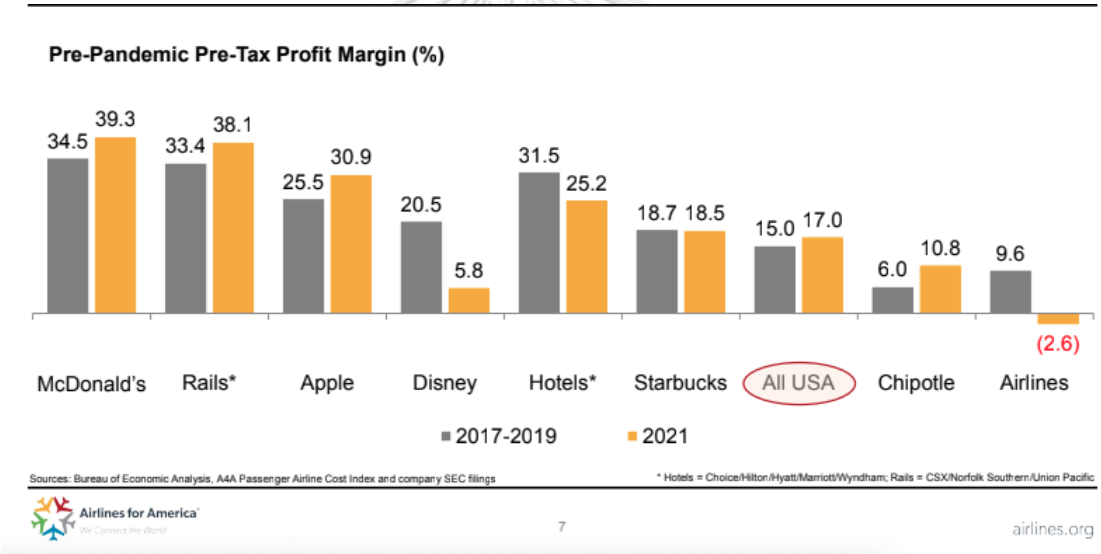


Figure 3: US Industries, 2021 Profitability (America, 2022)

Recently in 2022, the US has been affected by the Ukraine and Russia war and inflation. Consequently, the fuel price increased significantly and is anticipated to hit the all-time-high in 2022. According to Airlines for America, as referred to Figure 4, jet fuel spot price has increased 86% from 2019. The airlines have already spent 42 billion USD in 2022, which is a 125% increase from 2021.

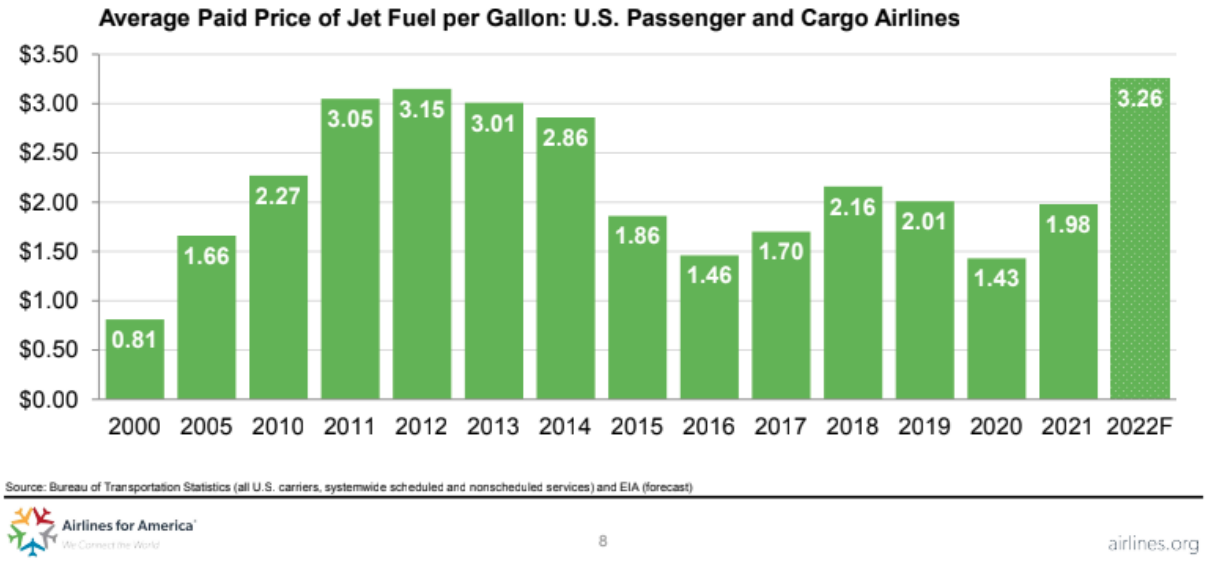


Figure 4: Jet Fuel Price for US Airlines (America, 2022)

Along with the fuel, airlines also have the labor, which is one of the 2 highest costs of the business. These 2 together account for 57% of the operating cost.

As mentioned above, the US airlines have been facing economic situations as well as financial distress for the past years. As a result, stakeholders of the airlines have taken action to solve some issues and maintain the business stability.

In response to Covid-19 pandemic, the US airline industry, incorporated with the Federal Aviation Administration (FAA), have issued some new changes to mitigate the spread of Covid-19 to employees and customers such as crew member medical certifications, guidance for airlines and airports, and some regulatory requirements. In terms of the strategies, airlines have come up with changes to strengthen the workforce pipeline, focus on profitable segments, and improve operational reliability such as increasing on-time arrivals.

1.2.1 US Aviation Traffic and Airspace

The United States is considered to be one of the busiest airspaces in the world as air travel is the preferred and fastest way of commuting. Over 16 million flights are handled by the Federal Aviation Administration (FAA, 2022a) yearly, over five thousand aircrafts in the sky are at peak operational times, and over 10 million are passenger flights yearly.

According to the FAA, which controls the national air space (NAS), the NAS is composed of 521 airport towers (263 Federal and 260 contract towers), 149 terminal radar control (TRACON) facilities (25 stand-alone and 124 combined ATCT), and 25 control centers (21 air route traffic control centers (ARTCC) and 4 combined control facilities (CCF). As referred to Figure 5 below, from airport tower to tower, the aircraft passes through multiple air traffic controls, each station also has extensive insights into weather information.

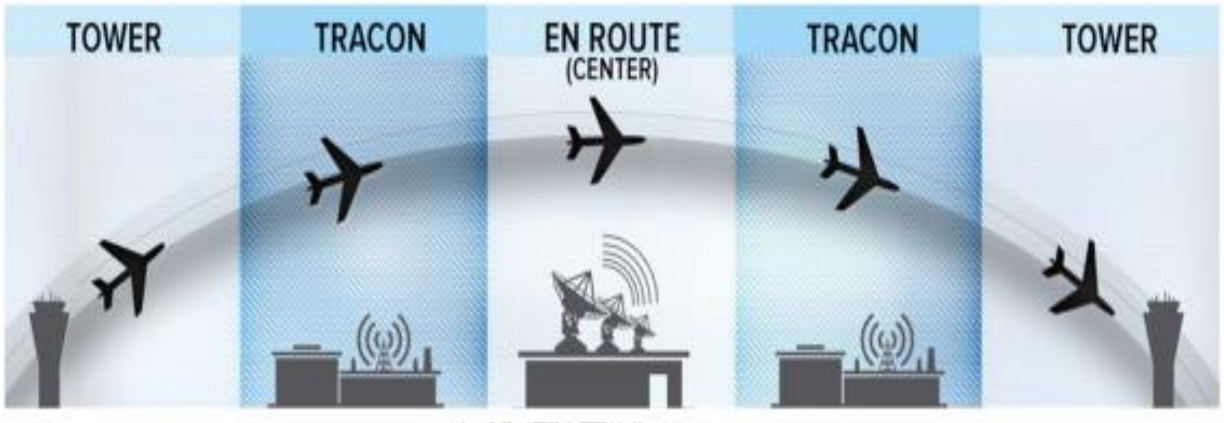


Figure 5: Air Traffic Control between Tower to Tower (FAA, 2022)

The United States separates the airspace in multiple ways such as altitude or even how busy each airport is. The different classes of airspace are Class A, Class B, Class C, Class D, Class E and Class G. Each of these airspaces has its own regulations based on what each class of airspace is intended for. Class A is only for IFR flight; the traffic is separated by the ground controller and the pilot is required to have an IFR license. Class B airspace is for the busiest airport there are 37 class B airports in the USA. There are flights from Class B airspace to Class A then landing at an airport with Class B airspace. The whole flight is controlled by the air traffic controller. The pilot does not have much control over which routes they can take. The Airspace Classification has been illustrated in Figure 6.

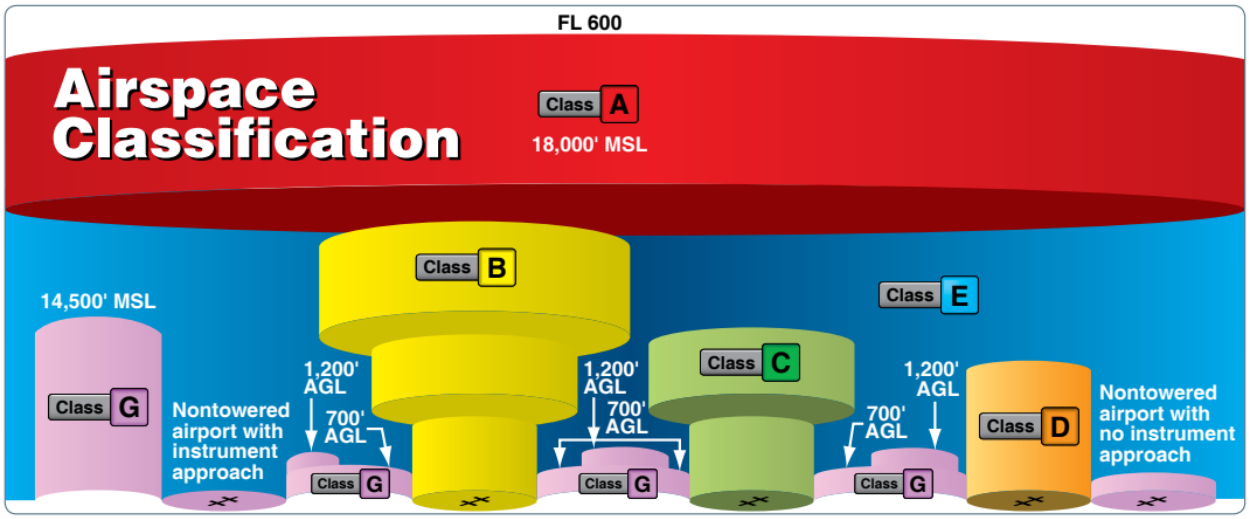


Figure 6: Airspace Classification (FAA)

1.2.2 US Aviation Infrastructure and Digital

The United States aviation infrastructure has tremendous coverage both physical and GPS coordinates. The infrastructure is the backbone, equally to highway systems. Airplane needs a way to know where they are in the sky relative to the ground. The VHF Omnidirectional Radio Range (VOR) and Distance Measuring Equipment (DME) provide the basic information that the plane needs to navigate. In terms of the digital system, the United States also uses GPS navigation, which could be interrupted or unavailable at any given time. To counter this GPS signal problem, there are 38 Wide Area Reference Stations (WRS), 3 Wide Area Master Stations, 3 GEO Satellites, and 6 GEO uplink subsystems. Each flight that flies under the Instrument Flight Rules (IFR) will have to follow the road in the sky to reach the destination. Each road in the sky will lead to how the aircraft conduct their final approach to the final destination. It sometimes takes the aircraft on the detour path rather than giving them a direct or shortest route. Also, the availability of radio navigation systems determines the route the aircraft will take, but with the help of GPS navigation, the aircraft can fly a more direct route rather than flying the traditional route. Or the pilot could choose the hybrid route that combines the traditional radio navigation with the GPS navigation. Instrument Flight Rule Chart can be shown in Figure 7.

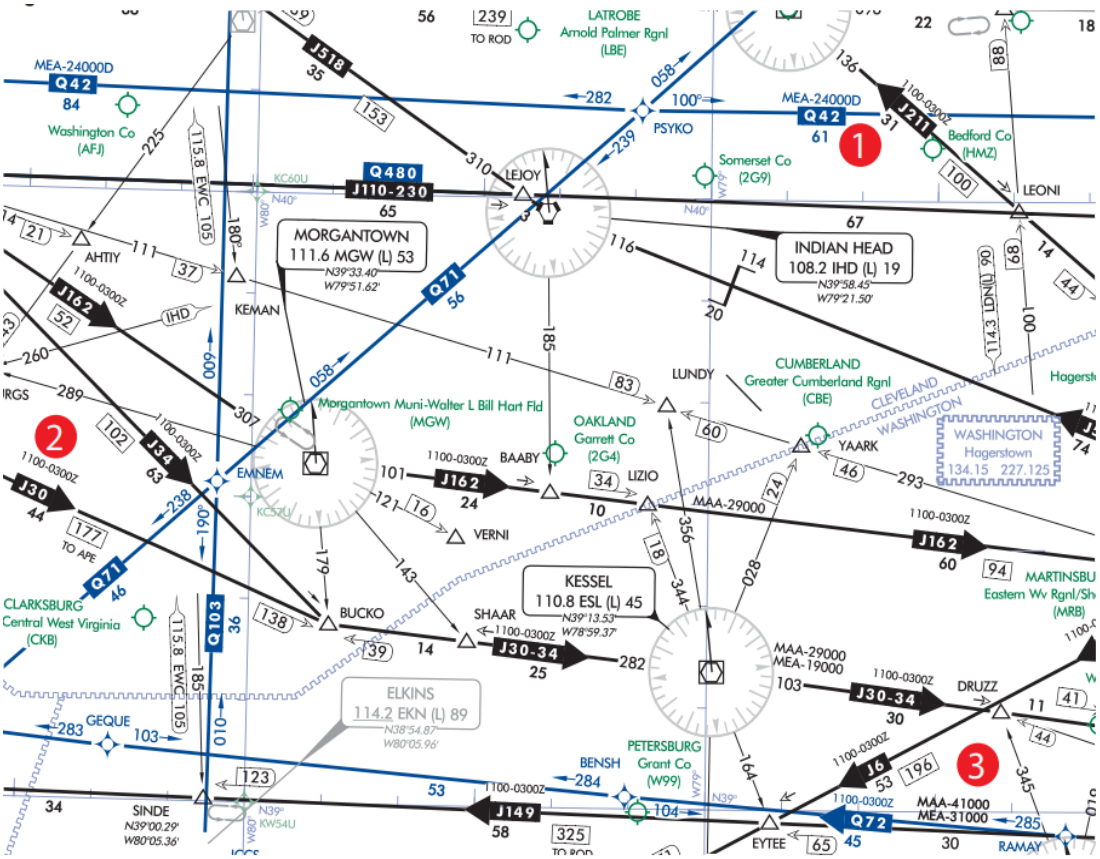


Figure 7: Instrument Flight Rule Flight Chart (FAA, 2023)

1.3 Aircraft Delays

Flight delays cause a ripple effect to the airline logistical management. From weather delays to missed connections and flight cancellations. Delays cause impact on the logistical support of the airline both on the ground and in the air. The ability to understand what will happen if the weather conditions are predicted to cause delays or even affect the flight will benefit the passengers and, especially, the airlines.

Each of the delayed flights induces a series of subsequent events. For example, the terminal gate might not be available, passengers might miss their connecting flights, subsequent aircraft delays, or even flight cancellations. The delay is unpredictable and nobody knows how long it will take or even to recover the scheduled flight. This causes the aircraft operator more time and resources to recover from the situation.

The aircraft delay causes setbacks and increases the unnecessary load to the air traffic controller since the aircraft is already airborne and eventually the plane needs to get back on the ground. Each airplane's time in the air is limited by how much fuel it has. The more fuel, the more time, but it also increases the unnecessary weight and the cost of operating the flight. Each flight carries an optimal amount of fuel with necessary spare. If the plane runs out of fuel, they could potentially cause more problems and become an emergency aircraft itself.

1.3.1 Factors that cause Aircraft Delays

According to the Federal Aviation Administration (FAA), the 5 factors that cause aircraft delays are as follows:

1. Carrier Delay

   Carrier delay is caused by activities in the air carrier. These include aircraft cleaning, aircraft damage, awaiting the arrival of connecting passengers or crew, baggage, bird strike, cargo loading, catering, computer, outage-carrier equipment, crew legality (pilot or attendant rest), damage by hazardous goods, engineering inspection, fueling, handling disabled passengers, late crew, lavatory servicing, maintenance, oversales, potable water servicing, removal of unruly passenger, slow boarding or seating, stowing carry-on baggage, weight and balance delays.

2. Late Arrival Delay

   Late arrival delay is caused by the late arrival of the same aircraft at a previous airport. An earlier delay at downstream airports is referred to as delay propagation. (FAA)

3. NAS Delay

   NAS Delay is the delay that is under the control of the National Airspace System (NAS). It includes non-extreme weather conditions, airport operations, heavy traffic volume, air traffic control, and etc.

4. Security Delay

   Security delay is caused by a security breach that results in evacuation of a terminal or concourse, reboarding of aircraft. It is also caused by

inoperative screening equipment and/or long lines in excess of 29 minutes at screening areas.

5. Weather Delay

Weather delay is caused by weather conditions that affect the operation, routing of the aircraft. Among the 5 factors, weather has the most influence on whether or not the aircraft is going to be delayed.

According to the National Airspace System (NAS), weather is the largest cause of aircraft delay. As referred to Figure 8, it accounts for 75.48% of the delays of greater than 15 minutes over the six years from June 2017 to May 2022.

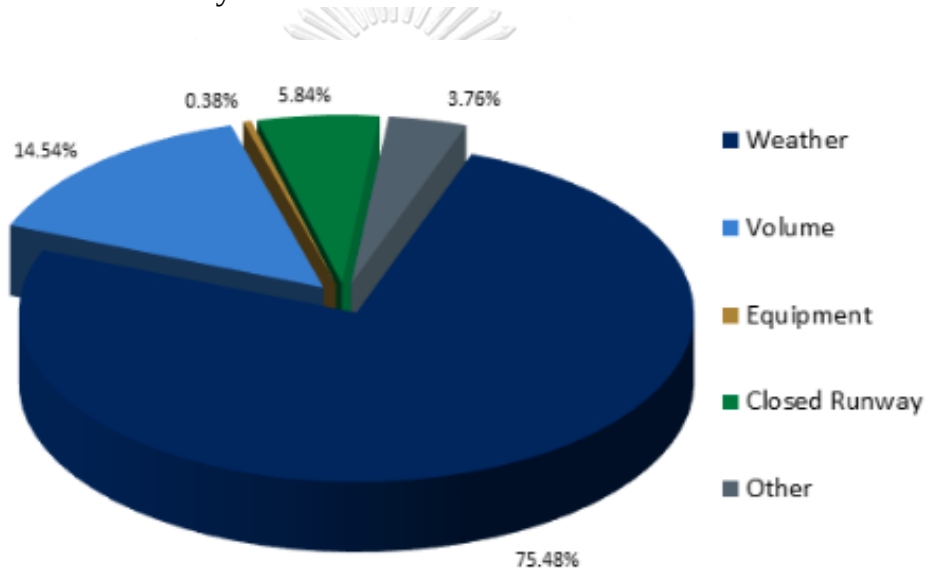


Figure 8: Cause of Air Traffic Delays in the National Airspace System (NAS) (FAA, 2022c)

In today's world, tropical storms, hurricanes, floods, and etc. pose huge challenges to the aviation industry. The weather plays an important role in aircraft delay. Weather is going to have more influence on aircraft delay in the future due to rapidly changing weather conditions in the World. Weather is something that has a pattern throughout the year and can be predictable at times. Other factors for aircraft delay can be prevented by using unlimited resources. For example, aircraft maintenance, if there is unlimited funding, the aircraft will have a lot more flying time than down time. Weather, on the other hand, is something that is observed and forecast. The shorter the forecast the more accurate it is.

At the same time, there are a number of weather conditions in different seasons that can cause flight delays such as hails, thunderstorms, microbursts, or even fog. Typical rains, snow, or the wind alone can also affect aircraft operations the same way. For example, a strong head wind is enough to result in a delay of the flight. A heavy fog that is near the ground level will prevent the pilot from landing the aircraft since the pilot cannot see the runway. Fog usually covers a vast area of land, but this weather phenomena is predictable. These weather conditions can be monitored in the Prog Chart in Figure 9. The airline could potentially have a backup plan for this airport in advance to prevent further delay.

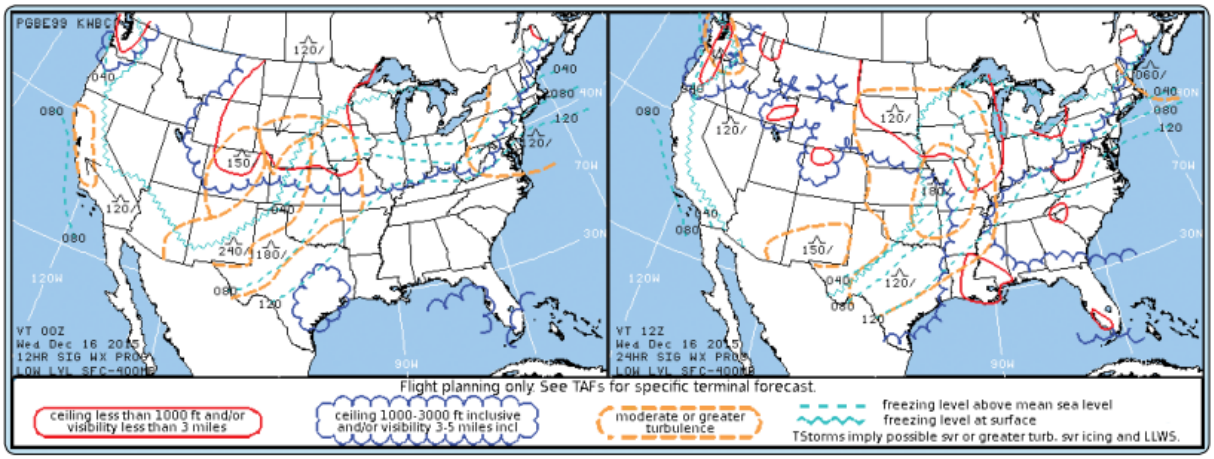


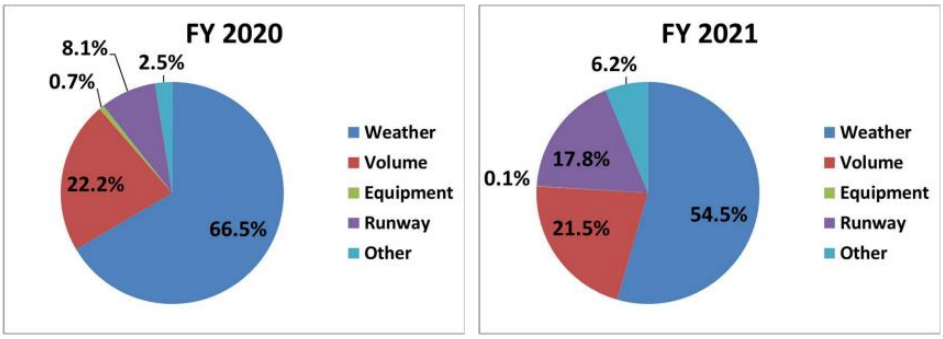
Figure 9: Prog Chart (FAA, 2022b)

These delays include delays that are caused by weather conditions at airports and enroute (Weather), FAA and non-FAA equipment malfunctions (Equipment), the traffic volume at an airport (Volume), runway capacity reduction (Runway), and other factors (Others). Flight delays below 15 minutes are not reported in OPSNET. ASPM reports the most dominant OPSNET delay cause for any flight with an ASQP Reported NAS Delay.

1.3.2 OPNETS Delay Cause

Operations Network (OPSNET) is the main source of National Airspace air traffic operations and delay data. The data that was collected is used to analyze the performance of the FAA's ATC facilities. According to the FAA, data collection delays to Instrument Flight Rules (IFR) traffic are set for 15 minutes or more from the arrival time, experienced by individual flights, which result from the ATC system detaining an aircraft at the gate, short of the runway, on

the runway, on a taxiway, and/or in a holding configuration anywhere en route. OPSNET data in Figure 10 shows that weather delay is the main cause of delay at US core airports. In 2021, the weather delay is less than in 2020, but still takes up the majority of the delay report.



Note: System impact delays are delays assigned to causal facilities in OPSNET and are composed of delays due to TMIs, departure delays, and airborne delays. System impact delays are also the basis for delays by class and delays by cause in OPSNET. (http://aspmhelp.faa.gov/index.php/OPSNET_Reports:_Definitions_of_Variables)

Figure 10: Sources of delays at Core 30 airport by type FY 2020 - 2021 (FAA, 2022c)

1.3.3 Current Aircraft Route Limitation

There are certain airport routes that are popular, for example, from New York to Boston, there are preferred routes for both high performance and normal performance aircrafts. The aircraft is preferred to fly a specific route that is agreed between the two airports. Most of the time, pilots do not have control over these routes, but to fly the preferred route. As the route is based on the traffic flow, the pilots cannot take a more direct route to the destination airport. This causes unwanted delays, which result in multiple setbacks and unnecessary workload to the system.

For civilian aircraft or general aviation aircraft, the preferred method is visual flight rules (VFR), which allows the pilot to fly without using the aircraft instruments. They operate by looking outside of the aircraft for visual reference or guidance of where they are going. Once the weather deteriorates, VFR is no longer possible. As a result, pilots with required skills will switch to fly IFR, which increases the load for the traffic controller.

Predicting the exact amount of aircraft arrival delay in minutes can be an incredibly challenging task due to a multitude of variables that significantly influence flight schedules. The diverse factors, such as the flexibility in pilot-

chosen routes, varying weather conditions at different altitudes, and the impact of headwinds and tailwinds, pose considerable obstacles to precise predictions.

The ability of pilots to select different routes based on real-time conditions and air traffic can lead to substantial deviations in flight durations. While some routes may encounter less congestion and more favorable weather conditions, others might face unexpected air traffic or adverse weather patterns, impacting the overall time taken for the journey.

Weather conditions, especially at different altitudes, play a crucial role in determining the actual flight duration. Variations in wind speed and direction, particularly headwinds and tailwinds, exert a significant influence on an aircraft's speed. Strong headwinds can slow down the aircraft, potentially causing delays, while tailwinds can provide a boost, reducing travel time.

The challenge in predicting these delays is compounded by the limitations of current datasets used in predictive models. While historical data forms the foundation for these models, they often lack real-time, granular information on dynamic weather changes, specific airspace conditions, or sudden operational issues that can significantly impact flight schedules.

Improving the accuracy of predictive models for aircraft arrival delays necessitates the integration of more comprehensive and real-time data. Incorporating live weather updates, detailed airspace conditions, and real-time aircraft performance metrics could enhance the predictive capabilities of these models. The intricate interplay of various variables within the aviation environment introduces complexities that are challenging to encapsulate fully in predictive models. The dynamic nature of aviation operations may continue to present hurdles in accurately predicting specific arrival delay times in minutes.

1.4 Problem Statement

The global airline industry has been experienced exponential growth, primarily propelled by economic expansion. Although demand for flights during the first hit of the Covid-19 dropped sharply, it bounced back as soon as the pandemic started to recover and is expected to surpass the number of pre-covid time. As the number of passengers and flights increase, the air traffic increases. This raises a question of how the industry is going to respond to this change.

The US airline industry, among the world’s largest, operates within a highly competitive landscape. Managing both high daily traffic and minimizing costs

to outpace competitors is imperative. Consequently, many airlines have implemented policy changes and new strategies to enhance operational efficiency. Some have launched new marketing campaigns, aiming to maximize profitability. However, persistent delays and cancellations remain significant issues, disrupting progress across various areas, from cost reduction to customer satisfaction.

To investigate the issue, there are many different types of delays as well as many different factors that cause delays. If there is a slight chance to predict whether the flight is going to be delayed and what the causes are, it will provide the logistical solutions that could be used to prevent further delays. Therefore, aircraft arrival prediction could potentially be a solution to improve the efficiency of the operation and the effectiveness of the contingency plans to respond to such events.

To understand more of the issues, machine learning is chosen to be a tool to extract insights on aircraft delays. Initially, available data is utilized while eliminating irrelevant information to predict future delays. This data undergoes classification through a learning method. The model is then tested using a separate set of data to gauge its accuracy. Any room for improvement, whether adjusting different features or incorporating additional data, leads to corresponding model adjustments. Consequently, the prediction model becomes capable of foreseeing potential aircraft delays.

The outcome of this prediction model empowers aircraft operators to adjust schedules and allocate sufficient buffer time. Seasonal and year-round schedule adjustments can be made, potentially enabling airlines to save substantial amounts while providing sustainable and predictable logistical support both on the ground and in the air.

The result will not only benefit the airlines, but also help increase the awareness of customers of the airlines as well as equip the workforce with a better tool to prepare for uncertainties efficiently and effectively. Moreover, with the current crisis of global warming, we can expect to see more severe weather issues that cause flight delays and cancellations. At the same time, the delay itself is the cause of more carbon emission. Reducing the downtime and increasing the operation efficiency would also be a sustainable solution and could potentially contribute to the better environment.

1.5 Objectives and Scope

Objectives:

1. To analyze the root causes of flight delays.
2. To develop a prediction model for aircraft arrival delay using machine learning methods over a period of time.

Scope:

1. Delay is based on the arrival time of the aircraft
2. Delay is defined as delay of 15 minutes or more after the scheduled arrival time.
3. The data used in this thesis is based on the US aviation database collected from January 2019 to March 2020.
4. This thesis focuses on 4 classification algorithms: Random Forest, AdaBoost, CatBoost, and Gradient Boosting.
5. Conduct an in-depth analysis on the optimal model to elaborate its performance characteristics.

1.6 Benefits

For Customers:

1. A better understanding of causes and what to expect from delays
2. An ability to prepare for delays once they are confirmed, change plans, and avoid schedule conflicts.
3. An ability to purchase travel insurance for compensation prior to the events

For Aircraft Operators:

1. Increasing efficiency for flight scheduling in different time throughout the year.
2. Ensuring airplane continuity with schedule adjustments and plane preparation
3. Improving customer relationships by offering reliable and trustworthy flight schedule
4. Reducing costs such as operational cost, fuel cost.

For Insurance Companies:

1. Appropriately adjusting terms and agreements to be more beneficial to stakeholders
2. Creating traveling insurance promotions that show confidence in providing necessary help to customers

For Environment:

1. Reducing carbon footprints from decreased downtime and increased efficiency in the operation

1.7 Research Timeline

The timeline of the research can be found in Table 1.

Table 1 Research Timeline

	Jul 22	Aug 22	Sep 22	Oct 22	Nov 22	Dec 22	Jan 23	Feb 23	Mar 23	Apr 23	May 23	Jun 23	Jul 23	Aug 23	Sep 23	Oct 23	Nov 23
Literature Review																	
Preliminary Analysis																	
Data Collection																	
Preliminary Findings																	
Proposal Paper and Deck																	
Aircraft Delay Prediction Model																	
Results and Conclusion																	
Defense Paper and Deck																	
Defense Submission																	

Chapter 2: Literature Reviews

The purpose of this chapter is to research and study existing papers and theories related to aircraft arrival delays.

2.1 Existing Research

2.1.1 Airline Schedule Buffer

Since the historic growth in air travel, a lot of problems have come up for passengers and airlines throughout the world. Flight time between two airports has been influenced by many random factors, such as weather, mechanical issues, and aviation congestion. Airlines have been scheduling their flights using a schedule buffer.

The schedule buffer includes both flight buffer and ground buffers. A buffer is the extra amount of time added to the minimum feasible flight or ground time to get the scheduled flight. Flight buffers reduce the chance of an aircraft being delayed. Also, to mitigate the delay propagation. (Brueckner et al., 2021a)

The current study provides the flight delay data as a statistical number and buffer estimation. As aircraft delay affects the whole schedule of flying and propagates throughout the schedule. (Brueckner et al., 2021a)

Currently, the on-time performance can be realized through the amount of buffer time used for each flight segment. (Wu, 2005)

The buffer time consists of ground buffer times and airborne buffer time. The smaller the buffer, while maintaining the airline on time performance is the ultimate goal of the airline. By adding time more than the minimum necessary in and between flights, unexpected delays can be absorbed and the effect of the delay to other flights can be mitigated or avoided (Nabin, 2016).

For an idealistic world the buffer times will be able to absorb all of the delays. The buffer time is there to control small delays, which results in maximizing fleet utilization (Wu, 2005).

Airline schedules buffers into their schedule by estimating the amount of time an aircraft takes to travel and adding the delay factor to the total time in order to compensate for lost time. Airlines have also adjusted their resources both human and property management to overcome flight delay.

However, airline schedule buffers come with a lot of problems. These problems are:

1. Over buffering: the schedule has over enough time for the aircraft to arrive on time. This leads to aircraft waiting to get into the gate or ramp.
2. Under buffering: the aircraft still remains in delay condition.
3. Lack of aircraft utilization: the aircraft is not being used enough and spends a lot of time staying on the ground.
4. Higher carbon footprint: The aircraft engine has to remain on for a longer period of time while waiting to be parked at its location.

These problems will be analyzed in Chapter 3: Methodology. However, this paper also investigated other methods: Aviation Infrastructure and Human Resource Investment and Aircraft Delay Prediction Model.

2.1.2 Aviation Infrastructure and Human Resource Investment

As we all know the aviation infrastructure is limited due to the number of airports and logistical supply chain of the system. Money is one of the most important driving factors of the aviation infrastructure supply chain.

Payment to airline crew is also determined in part by the length of scheduled flight, and by increasing the buffer time it increases the crew expense (Nabin, 2016). Next is if the aircraft doesn't need the buffer, then aircraft might have further problems with gate or ramp availability, which increase the airline operating cost (Nabin, 2016). These factors increase the carbon footprint of the airline and create global warming gas. There are more and more passengers who choose air travel as demand increases. The current system will be handling more and more load. The system will be prone to an increased number of flight delays. If more investment has been made to the aviation infrastructure it will be able to handle more load and be more prepared. Using comparative static analysis shows capacity constraint suppresses demand, reduces flight frequency and increases passenger cost. (Zou and Hansen, 2012)

However, the challenges for this method are the following:

1. There is no unlimited amount of money.
2. The equipment or manpower in reserve will not all be used.
3. The airlines will be over paying for their provided service.
4. The airlines will need to build awareness about the resource that was used in the system to build a precise schedule with absolutely minimum delay for the passengers.

This method could be optimized with the prediction model and will be explained further in Chapter 3: Methodology.

2.1.3 Aircraft Delay Prediction Model

An Aircraft Delay Prediction Model is a machine learning model that uses different methods and aviation data to predict the likelihood of an aircraft experiencing delays. Some of the methods used to create this model include Random Forest, AdaBoost, Gradient Boosting, and decision trees.

The model is designed to analyze various factors that could potentially cause delays, such as weather conditions, air traffic congestion, technical problems, and crew availability. By processing this data, the model generates a prediction for the likelihood of an aircraft experiencing delays.

Over time, the model has been refined using different methods and data sources to improve its accuracy and F-1 score. This ongoing effort is crucial to ensure the model is effective in predicting delays and minimizing disruption to air travel.

One study conducted by Hu et al. (2021) used Random Forest to predict flight delays using data from Guangzhou Baiyun International Airport. They defined delay as an aircraft arriving five minutes or more after the scheduled arrival time. The study found that the optimal parameters for the Random Forest model were 50 trees, optimal leaf size of 5, and the minimum mean square error of 0.1096. The study achieved an accuracy rate of 66%, which was the most accurate and least amount of error compared to other models.

Liu et al. (2020) conducted a study on generalized flight delay prediction using Gradient Boosted Decision Trees. Their data was collected from the Civil Aviation Administration of China (CAAC), and they defined delay as a 15-minute delay in arrival time. The study found that the Random Forest model outperformed other models with an accuracy rate of 78.02%. They also focused on weather data, including weather conditions at the departure and arrival airports, wind direction, and wind power. The study found that the Gradient Boosted Decision Tree method provided the highest accuracy rate of 87.72%.

Gui et al. (2020) used Automatic Dependent Surveillance-Broadcast (ADS-B) data in their study to predict flight delays. The ADS-B data contains flight data such as ICAO identity number, position, and velocity. The study also used weather information such as wind direction, wind speed, and weather conditions. They used Random Forest-based and Long Short-Term Memory (LSTM)-based architectures to predict flight delays. The Random Forest-based

method achieved an accuracy rate of approximately 90% for binary classification and was able to overcome the overfitting problem.

In another study, Khaksar and Sheikholeslami (2017) analyzed flight delays from US and Iranian networks using decision trees, Random Forest, Bayesian classification, K-means clustering, and hybrid approaches. The study found that the data from different countries can be related to each other, and the prediction accuracy in the US was slightly lower than that in Iran.

In summary, various methods and data have been used to develop aircraft delay prediction models. These models have shown improvement in accuracy and F-1 scores, and they continue to be a valuable tool for airlines to manage their operations and improve customer satisfaction.

Based on the information provided above, here are some potential challenges of aircraft delay prediction using machine learning methods:

1. Data quality and quantity: One of the biggest challenges in developing accurate machine learning models for predicting aircraft delays is having access to high-quality and sufficient amounts of data. The quality of data can vary depending on the source and method of collection, and missing or inaccurate data can affect the accuracy of the model. Additionally, the amount of data needed to develop accurate models can be quite large, and collecting and processing this data can be time-consuming and resource-intensive.
2. Feature selection: Another challenge is selecting the most relevant features or variables to use in the model. There are many factors that can contribute to aircraft delays, including weather conditions, air traffic congestion, mechanical issues, and human error. Identifying the most important factors to include in the model can be difficult, and choosing the wrong features can lead to inaccurate predictions.
3. Model selection and tuning: There are many different machine learning algorithms and techniques that can be used for aircraft delay prediction, and selecting the best approach for a particular dataset can be challenging. Additionally, properly tuning the model parameters to optimize its performance can be time-consuming and require extensive testing.
4. Dynamic nature of data: The factors that contribute to aircraft delays can vary over time and in different locations. This can make it difficult to develop accurate and reliable models that can be applied to different situations. Additionally, the model may need to be updated regularly to account for changes in the underlying data.

5. Interpretability: Another challenge of machine learning models is interpretability. It can be difficult to understand how a model arrived at a particular prediction, which can make it difficult to identify and address any biases or errors in the model. This can be particularly important in the context of aircraft delay prediction, where accurate and transparent models are essential for ensuring passenger safety and minimizing disruptions to air travel.

These insights will be applied to the prediction model, which will be thoroughly explained in Chapter 3.

2.2 Existing Research Comparison

Different airlines have their ways of dealing with delayed aircrafts. Airlines have been overestimating their flight time to make sure that they are on time. This is the current solution that most airlines are doing to compensate for the lost time. The airlines will want to have a flight delay prediction model to improve their schedule, build trust and increase efficiency.

A traditional way is to simply focus on flight time equal to distance divided by time. After the time is calculated then the delay factor is added on to the total time. If the aircraft is delayed often then the scheduler will then make an adjustment to the schedule to accommodate the longer commute time. Nevertheless, the airline does not have control over the weather and its effect on aircraft delay. Therefore, a lot of weather data and weather prediction models are available.

Aircraft scheduling buffer has been a widely used method to compensate for aircraft flight delay. It has been used to add more time to the schedule for multiple types of delay. Brueckner et al. (2021) utilized the method of airline scheduling buffer choice and study the shocks influencing flight times.

Brueckner et al. (2021) provided an insight on the theoretical model to analyze the airline's choice of buffering method, and also suggest that a mitigation of delay propagation can be recovered entirely by the ground buffer and the second is flight buffer.

Hajko and Badánik (2020) suggested both benefits and the drawback of the airline scheduling buffer, while using artificial data instead of actual flight data. These specific components caused delays in passenger and baggage, cargo and mail, aircraft and ramp handling, technical and aircraft equipment, damage to aircraft & automated equipment failure, flight operations and crewing and other airline related causes. As the author mentioned, weather is not the reason for every flight delay, but when they do the delay is significant.

Another method that is not widely used is to upgrade the aviation infrastructure. Zou and Hansen (2012) suggested this method and discovered that there is a balance in the complicated set of adjustment between passenger demand, air fare, flight frequency, aircraft size, and flight delays which will lead to an equilibrium. Currently there are so many aircraft that can land at the same time, while building another runway or an airport will increase the capacity of the aviation system.

Flight delays have been a significant factor at each airport. It has a huge impact on the economic cost and consequences. Federal Aviation Authorities and a lot of research has been conducted to measure aviation delay and its economic impact. On the other hand, there are a lot of reports of how much weather affects the environmental cost is widely available Dissanayaka, (2019).

It is significant that each delay must be stopped before it starts to propagate and affect other flights. It is apparent that each aircraft delay causes further delays in the schedule. Schedule buffer has its downside as well. Buffer makes the flight schedule and turnaround time longer than necessary. It also reduces the utilization of the aircraft and drives the operating cost higher.

Flight delay prediction has been the topic of multiple studies in the past. In 2019, Khaksar and Sheikholeslami (2019) had conducted a study between the aviation data, based of US and Iran through multiple prediction models. They found that the prediction model could be applied to different countries or locations and the prediction model would predict the outcome just fine although there are different reasons for aircraft delays for both of the locations.

Random Forest method is widely used to predict aircraft delay. Gui (2020) has applied Random Forest method to his ADS-B data from China as well. There are multiple prediction methods that Gui used, but the best accuracy model was the Random Forest model. Liu (2020) has compared multiple prediction methods including Gradient Boosting, K-nearest Neighbors, Support Vector Machine (SVM), and Random Forest method. The best predicting model for Liu was Gradient Boosted decision tree method. Another paper from Hu (2020) also affirms that the method of Random Forest is used to predict flight delay as well. Therefore, Random Forest should be set as a baseline for future study and comparison with other prediction models.

Since the world is going to more personalize traveling, smaller aircraft are preferred. More and more smaller body aircraft are in demand. On the other hand, the aviation infrastructure has limited resources, and larger aircraft is preferred. This thesis would not focus on aircraft maintenance, which is mostly predictable and prepared for as well as the load on the aviation infrastructure

that could be calculated. Therefore, this thesis would not focus on increasing investment to the business of aviation, but would focus on aircraft delay predictions.

At the end of the day, it all comes down to the money and the breakeven point for the airline. The industry is open to finding a possible solution and cure to each specific type of delay. The accurate model will save a lot of money for the airline.

A summary of the comparison of the journal articles is shown in Table 2.

Table 2 Comparison with other journal articles

Reference	Methodology	Objective
(Mueller and Chatterji)	Post Operation Evaluation Tool (POET) using delay time probability functions	Acquire the right distribution model for aircraft delay schedule
(Wu, 2005)	Markov Chain algorithm:	Schedule reliability buffer
(Zou and Hansen, 2012)	Comparative static	Aviation infrastructure investment
(Zou and Hansen, 2012)	Utilizing the supply and demand equilibrium to understand the requirement of the system.	Aviation capacity investment
(Kafle and Zou, 2016)	Newly formed delay at node refers to delay that occurs between node and its immediate upstream node.	Additional times than the minimum necessary in and between flights
(Kasirzadeh et al., 2017)	Airline Crew Scheduling: Models, Algorithms,	Aircrew scheduling deconflict using column generation
(Khaksar and Sheikholeslami, 2019)	Using information from the US and Iran to find the best method of prediction. The prediction model is then applied to both country aviation data.	Predict aircraft delay using Decision Tree, Random Forest, Bayesian classification, K-means clustering and hybrid approach

Table 2 Comparison with other journal articles (continued)

Reference	Methodology	Objective
(Dissanayaka et al., 2019)	Evaluate the impact of departure flight delays at ground operations with International Civil Aviation Organization (ICAO) engine emission databank	Evaluation of emissions from delayed departure flights
(Hajko and Badánik, 2020)	Analyzing the impact of the schedule buffer on various operational factors cost, aircraft usage etc.	Implement schedule buffer into the schedule
(Gui et al., 2020)	Collecting data from automatic dependent surveillance broadcast (ADS-B) data including basic weather information and conditions.	Using Random Forest-based and LSTM-based architectures to implement the prediction of flight delays
(Liu et al., 2020)	Using data from Civil aviation administration of China (CAAC) to predict flight delay using Gradient Boosting and decision trees.	Using Gradient Boosting, Decision Tree, K-nearest Neighbors, Support Vector Machine, Naive Bayes Classifier and Random Forest method
(Brueckner et al., 2021b)	Random shocks influencing flight times are discrete rather than continuous	Schedule buffer to prevent delays
(Hu et al., 2021)	Research of flight delay prediction based on Random Forest method	Using Random Forest to predict flight delays
(Brueckner et al., 2021a)	Analyzing different choices for single and two flight models.	Schedule buffer choosing which method to mitigate the propagated delay

By using the flight data to build a prediction model, this thesis expects to create benefits to the aircraft operators, passengers, and all the stakeholders as mentioned in 1.6 Benefits

2.3 Related Theories

Machine learning is a subfield of artificial intelligence that focuses on developing algorithms and models capable of learning and making predictions or decisions without being explicitly programmed. It involves the use of

statistical techniques to enable computers to identify patterns, extract knowledge, and continuously improve their performance based on data. One of the key concepts in machine learning is supervised learning, where algorithms are trained on labeled data to make predictions or classifications.

There are mainly two types of supervised machine learning algorithms: Classification, which utilizes AI to group different objects (values) into categories, known for tasks such as photo identification, and Regression, which predicts an output value based on a set of features and is commonly used to forecast prices or values of objects. Within the realm of supervised learning, tree-based machine learning algorithms have emerged as highly popular choices for classification tasks. In the case of aircraft arrival delay prediction, the most suitable algorithm falls under the supervised classification machine learning algorithms category.

Tree-based machine learning algorithms, including decision trees, Random Forest, and gradient boosting, have gained widespread popularity for a variety of compelling reasons. Firstly, decision trees are inherently interpretable and mirror human decision-making processes by creating a tree-like structure of logical, easy-to-follow rules. This interpretability is vital in applications where understanding and explaining the decision-making process is crucial, such as in medical diagnoses, credit risk assessments, or legal proceedings.

Random Forest, an ensemble of decision trees, offers a substantial improvement in predictive accuracy by combining the individual strength of multiple trees. It mitigates the risk of overfitting, provided robustness against noisy or irrelevant features, and can handle mixed data types effectively. The ability to handle missing values and outliers adds to its appeal, making it well-suited for complex datasets with a high number of features. Moreover, Random Forest provided feature importance, aiding in feature selection and understanding the data's underlying patterns.

Gradient boosting algorithms, such as XGBoost, LightGBM, and CatBoost, have also gained popularity due to their outstanding predictive performance. They iteratively enhance the model's accuracy by correcting the errors of previous models, making them highly effective in predictive tasks where precision is paramount. These algorithms excel in both classification and regression problems and offer impressive flexibility through parameter tuning, allowing data scientists to fine-tune the model's behavior for specific tasks.

Tree-based algorithms exhibit robustness against outliers and imbalanced datasets, reducing the need for extensive data pre-processing. They can handle mixed data types, including numerical and categorical features, simplifying the

workflow and making them accessible to a broader range of users. Furthermore, they offer strong support for dealing with missing data, reducing the necessity for data imputation. Their adaptability to various data conditions and their capacity to tackle violations of assumptions make them suitable for a wide array of applications, including finance, marketing, healthcare, and natural language processing.

The popularity of tree-based machine learning algorithms in the context of classification tasks can be attributed to their interpretability, versatility, robustness, predictive power, and adaptability to various data scenarios. These models continue to be the preferred choice for many data scientists and machine learning practitioners, offering a powerful toolset for solving a diverse range of real-world problems. While decision trees may be easier to compute and interpret, their susceptibility to overfitting makes Random Forest an attractive choice. The inclusion of additional features may be necessary to enhance prediction accuracy. On the other hand, AdaBoost, with its boosting methodology, offers simplicity and a shorter runtime, making it a valuable contender for predictive tasks where a straightforward approach is desired.

2.3.1 Random Forest Classification

Random Forest is a popular and commonly used algorithm. Random Forest is a supervised machine learning algorithm that is used with classification and regression problems. It can handle the data set containing continuous variables. The reason of using the first two methods of Random Forest and AdaBoost is to determine whether using the full complex forest is different than using a stump. The ideology is to start with smaller model combining multiple models together.

In the Random Forest method, the way that the model thinks are parallel. Random Forest method includes diversity and stability. Each tree in the Random Forest model is created independently, out of different data and attributes. The result is based on majority averaging. The Random Forest Classifier, which can be seen in Figure 11, was chosen as the estimator in the data.

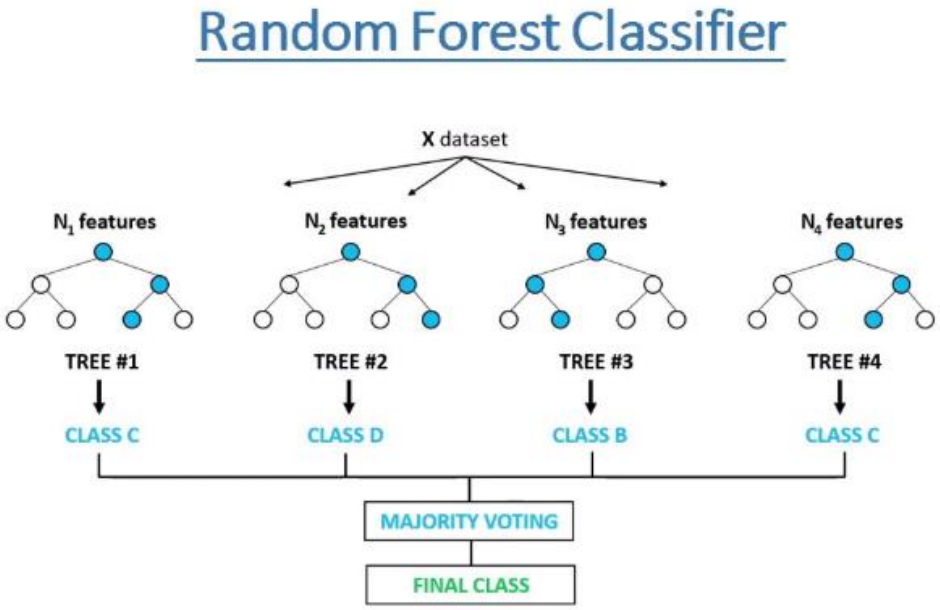


Figure 11: Random Forest Classifier (FreeCodeCamp, 2020)

Equation (1) presents the classification outcome when values of 0,1…,K-1 for m be the proportion of class k observations in node m. If m is a terminal node, predict_proba for this region is set to p_{mk}. Common measures of impurity are with Equation (2). The Entropy uses the Shanon Entropy as a tree node to split different criterion which is also equal to minimizing the log loss. This is shown in Equation (3).

$$p_{mk} = \frac{1}{n_m} \sum_{y \epsilon Q_m} I(y = k) \quad (1)$$

Gini Impurity Equation:

$$H(Q_m) = \sum_k p_{mk}(1 - p_{mk}) \quad (2)$$

Log Loss or Entropy Equation:

$$H(Q_m) = -\sum_k p_{mk} log(p_{mk}) \quad (3)$$

As mentioned before, Random Forest is used as a supervised learning algorithm method, which can be used for classification and regression. The trees are the subsection of the forest. For a forest with a lot of trees, it becomes

richer. The Random Forest creates a decision tree on randomly selected data. This function gets predictions from each tree and produces the best solution. Also, it produces a good indicator of the importance of the data. The advantage of Random Forest is that it takes all of the information and averages out the prediction, which cancels out the bias.

The Random Forest algorithm offers a range of hyperparameters that allow for fine-tuning and customization of the model to suit the specific needs of a machine learning task. Some of the other hyperparameters that were used:

'min_weight_fraction_leaf': This parameter sets the minimum weighted fraction of the total sum of weights (of all input samples) required to be at a leaf node. It can be used to control the minimum amount of data required for a node to be created as a leaf.

'max_leaf_nodes': When set to a non-None value, this parameter limits the maximum number of leaf nodes in a tree, effectively controlling the depth and complexity of each decision tree in the ensemble.

'min_impurity_decrease': It specifies a threshold for impurity decrease, where a split is considered only if it results in a decrease in impurity above this threshold. This parameter can be used to regulate the splits made during tree construction.

'bootstrap': If set to True, bootstrapping is enabled, meaning that the algorithm will use random resampling with replacement when constructing individual trees. Bootstrapping can increase the diversity of the ensemble.

'oob_score': When set to True, this parameter enables out-of-bag (OOB) scoring. OOB samples are used to estimate the model's generalization accuracy, providing a useful metric for assessing the model's performance.

'n_jobs': This parameter allows you to specify the number of CPU cores to utilize during model fitting. Parallel processing can significantly speed up the training process, especially for large datasets.

'random_state': By setting this parameter to a fixed value (an integer), you can ensure reproducibility in model training. It seeds the random number generator, making your results consistent across different runs.

'verbose': This parameter controls the verbosity of the model, where higher values result in more detailed output during training. Setting it to 0 typically means minimal output, while higher values provide more information.

'warm_start': If set to True, this parameter allows you to reuse the existing solution and continue training the Random Forest. It can be beneficial when you want to incrementally update the model with new data.

'class_weight': This parameter addresses class imbalances by assigning different weights to classes. When set to "balanced," the algorithm automatically adjusts class weights based on their frequencies in the training data.

'ccp_alpha': It controls the complexity of the decision trees by setting a non-negative value for the cost-complexity pruning (CCP) parameter. Higher values lead to more pruning, simplifying the trees.

'max_samples': This parameter limits the number of samples used for training each tree, which can be useful for controlling the composition of the ensemble, especially when working with large datasets.

2.3.2 AdaBoost Classification

AdaBoost is a method that uses the completed training dataset to train. It is a sequential process. Each of the following models tries to correct the errors of the previous models. The model tries to correct the errors so, for the next iteration, it is better as shown in Figure 12. The stump is a single unit or iteration of AdaBoost. A stump is a decision tree with only single split.

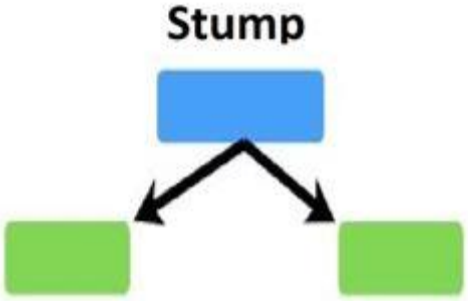


Figure 12: Single Unit of AdaBoost Model. (Medium, 2019)

Each model is trained on the same dataset, but each of the data sample is assigned a different weighting factor in the previous model's success. It is the learning process in sequence. The weighing factor is then reassigned in every iteration to make a better classifier than the previous iteration. The process begins when a subset is selected from the original dataset. Then all training examples are assigned the same weight. A base model is then trained on this

subset. The final model of this subset will be used to make predictions on all of the data. The errors are calculated using the actual values and the predicted values. Each sample has a sample weight that will determine how much it will contribute to the final outcome of the model.

The AdaBoost classification offers a wide range of hyperparameters that allow for fine-tuning and customization of the model to fit the specific needs of a machine learning task. Some of the hyperparameters that were used:

n_estimators: The n_estimators hyperparameter controls the number of weak learners (base estimators) to be used in the ensemble. Adjusting this variable allows you to find the right balance between model complexity and performance. Increasing the number of estimators can lead to better accuracy, but it may also increase the risk of overfitting. Conversely, using too few estimators may result in underfitting. Therefore, adjusting n_estimators is crucial to achieve the best trade-off between bias and variance in your model.

algorithm: The algorithm hyperparameter defines the boosting algorithm used by AdaBoost. The choice between 'SAMME' and 'SAMME.R' impacts the performance of the algorithm. 'SAMME.R' typically performs better than 'SAMME' and is recommended for multiclass classification problems. Adjusting this hyperparameter ensures that the algorithm aligns with the specific requirements of your task.

learning_rate: The learning_rate hyperparameter influences the contribution of each weak learner to the final prediction. A smaller learning rate makes the model more conservative and robust, while a larger learning rate can lead to overfitting. By adjusting this variable, you can control the impact of individual estimators on the ensemble's prediction. Smaller values are often favored when striving for a more reliable model.

base_estimator_criterion and base_estimator_splitter: These hyperparameters allow you to specify the criterion and splitting strategy for the base estimator, which is typically a decision tree. Adjusting these variables can tailor the underlying decision trees to the characteristics of your data. For example, you can use 'entropy' or 'gini' as the criterion to measure impurity differently or experiment with 'best' or 'random' as the splitting strategy. Fine-tuning these hyperparameters can help you create base estimators that capture data patterns more effectively.

2.3.3 CatBoost Classification

Category boost or CatBoost is an algorithm for Gradient Boosting on decision trees. A sample of CatBoost model is as shown in Figure 13. It is made as a classifier to deal with categorical features automatically. In every step, the leaves from the previous tree are split using the same condition. The feature-split pair that has the lowest loss is selected and used throughout the level's nodes. This balance tree helps improve the efficiency in the implementation, decrease the prediction time, make swifts model appliers, and control overfitting.

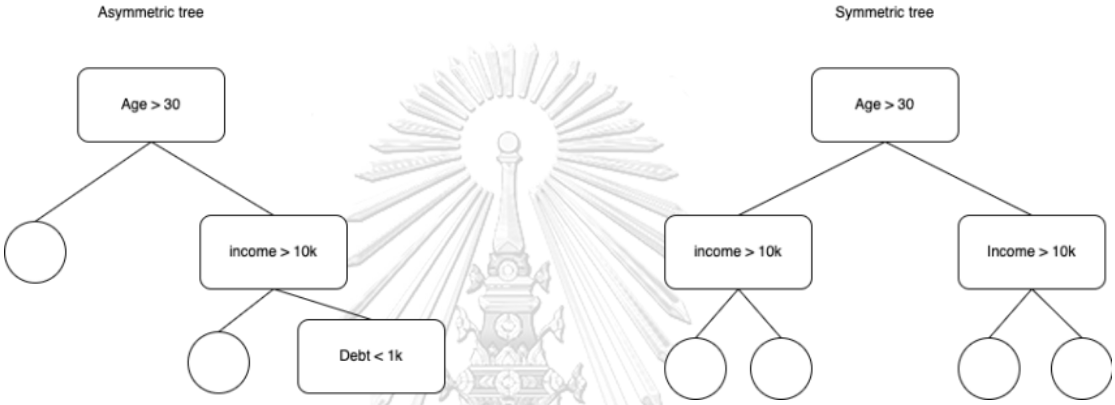


Figure 13: Category Boost Model (Toward Data Science, 2022).

Another feature of CatBoost is ordered boosting. The classic boosting can easily cause overfitting on small dataset due to prediction shift. CatBoost uses the concept of ordered boosting, which is another way to train the model on a smaller set of data while including the residuals on another set of data. This prevents overfitting.

Last feature of the CatBoost is native feature support. CatBoost supports all types of features. It can be numeric, categorical, or text, which saves time and effort of pre-processing. For example, the native feature support provides one-hot encoding, statistics based on category, search for combination, and ranking. This reduces the pre-processing time and limits the possible combinations to make a good model as well.

The CatBoost classification offers a smaller range of hyperparameters that allow for fine-tuning and customization of the model when compare to other classification and model to fit the specific needs of a machine learning task. Some of the hyperparameters that will have an effect on the accuracy that were used:

Iterations: This hyperparameter determines the number of boosting iterations or trees in the ensemble. A higher number of iterations can lead to better performance, but it can also increase the risk of overfitting, so it's essential to find the right balance.

Learning_rate: The learning rate controls the step size during the optimization process. It influences the impact of each tree on the final prediction. Smaller learning rates make the model more conservative and robust, while larger rates can lead to overfitting. Tuning this parameter helps you achieve the right trade-off between accuracy and generalization.

Depth: The depth hyperparameter defines the maximum depth of each decision tree in the ensemble. Deeper trees can capture complex relationships in the data but may also lead to overfitting. You can adjust this parameter to control the complexity of the model.

2.3.4 Gradient Boosting Classification

Gradient Boosting is an algorithm that builds an additive model in forward stage-wise fashion. A sample how each stage look is as shown in Figure 14. The model allows for optimization of arbitrary differentiable loss functions. Each stage is defined as n_classes_, which are regression trees that are fit on the negative gradient of the loss function. The parameters consist of loss, learning_rate, n_estimators, subsample, criterion, min_samples_split, min_samples_leaf, min_weight_fraction_leaf, max_depth, min_impurity_decrease, init, random_state, max_features, verbose, max_leaf_nodes, warm_start, validation_fraction, m_iter_no_change, tol, and ccp_alpha. The loss function is to be optimized. It is directly link to the logistic regression, which uses binomial and multinomial deviance.

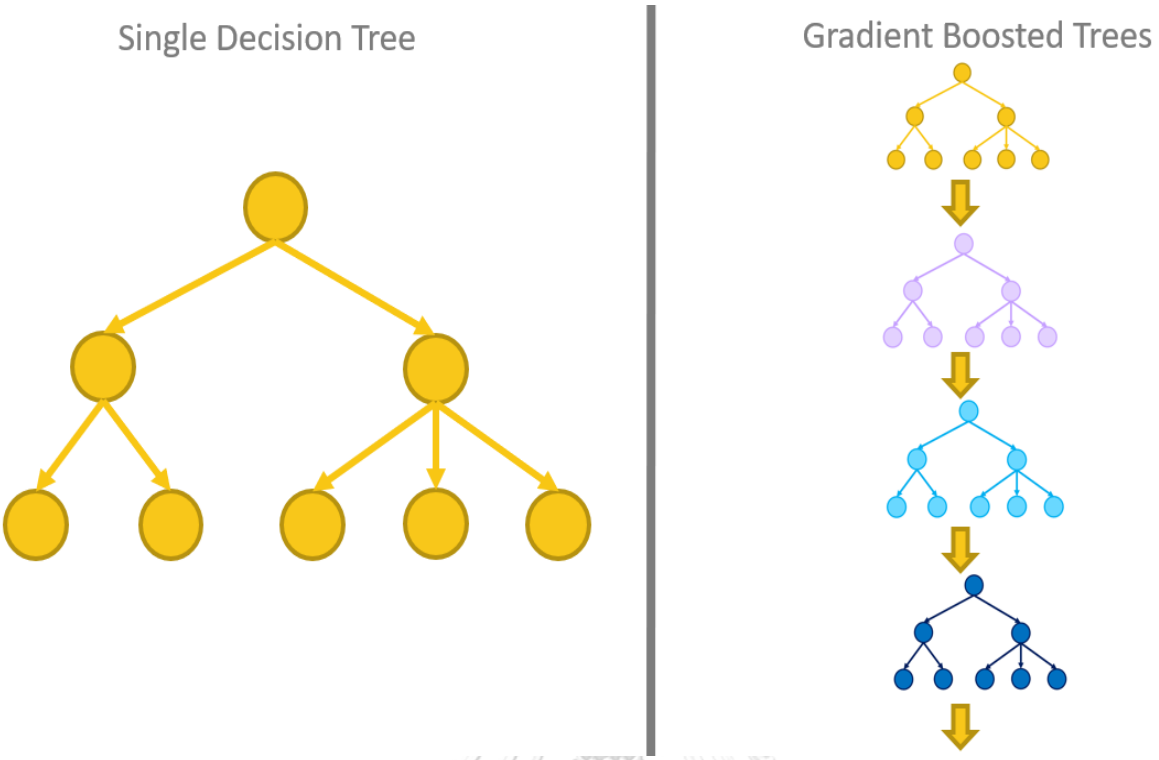


Figure 14: Gradient Boosting Model (Toward Data Science, 2019).

Gradient Boosting recovers the AdaBoost algorithm. Gradient Boosting is a powerful machine learning algorithm that combines multiple weak learners to create a strong predictive model. The Gradient Boosting model offers a wide range of hyperparameters that allow for fine-tuning and customization of the model to fit the specific needs of a machine learning task. Some of the hyperparameters that were used:

n_estimators: This hyperparameter controls the number of weak learners (base estimators) to be used in the ensemble. Increasing the number of estimators generally leads to better accuracy, but it can also increase the risk of overfitting. It's crucial to find the right balance between model complexity and performance by adjusting this parameter.

learning_rate: The learning rate influences the contribution of each weak learner to the final prediction. A smaller learning rate makes the model more conservative and robust, while a larger learning rate can lead to overfitting. Adjusting this variable allows you to control the impact of individual estimators on the ensemble's prediction.

min_samples_leaf: The min_samples_leaf hyperparameter controls the minimum number of samples required to be at a leaf node. Adjusting this parameter can impact the model's ability to capture fine-grained details in the data.

min_samples_split: The min_samples_split hyperparameter sets the minimum number of samples required to split an internal node. Adjusting this parameter can affect the model's sensitivity to small variations in the data.

max_depth: The max_depth hyperparameter controls the maximum depth of the individual decision trees used in the ensemble. Adjusting this parameter can prevent overfitting. Smaller values limit the tree's depth, making the model more robust, while larger values may result in overfitting.

max_features: This hyperparameter determines the number of features considered for splitting at each node. For example, setting it to "sqrt" means the square root of the total number of features is considered. Experimenting with different values for max_features can help fine-tune the model's performance.

criterion: The criterion hyperparameter defines the function to measure the quality of a split. Typically, options like 'friedman_mse' are used for gradient boosting, but you can experiment with other criteria such as 'mse' or 'mae' for regression tasks.

subsample: The subsample hyperparameter controls the fraction of samples used for fitting the weak learners. Setting it to a value less than 1.0 introduces randomness and can help reduce overfitting.

Fine-tuning these hyperparameters is essential to achieve the best balance between bias and variance in your Gradient Boosting model, ultimately leading to better predictive performance.

2.3.5 Tuning Hyperparameter

GridSearchCV is known for its comprehensiveness. It conducts through search through predefined hyperparameter combinations, ensuring that no potential configuration goes unexplored. This approach is valuable since every combination could be the absolute best hyperparameters for the model.

The simplicity and ease of use make GridSearchCV a good option. Machine learning user at all levels can readily understand and employ this method. It involves specifying a grid of hyperparameters to explore, and the tool takes care of the rest, making it accessible to both beginners and experienced data scientists.

While GridSearchCV has a lot of benefits, it might not always be the ideal choice. In cases where the hyperparameter space is extensive and exploring every combination is computationally impractical, other methods like RandomizedSearchCV, Bayesian optimization, or genetic algorithms can offer more efficient and resource-friendly alternatives. Similarly, when the relationships between hyperparameters are complex and intertwined, GridSearchCV may not effectively navigate the search space. More advanced methods can often better capture these intricate interactions.

GridSearchCV provides a reliable baseline for model performance. It can establish a benchmark for the achievable results with different hyperparameters. This baseline becomes a reference point against other method which can be use to compare the outcomes of more sophisticated hyperparameter tuning methods. GridSearchCV is a valuable and essential tool for hyperparameter tuning, especially when simplicity, thoroughness, and reproducibility is essential. It is just one piece of the hyperparameter optimization puzzle, and its utility depends on the specific characteristics of the dataset, the available computational resources, and the complexity hyperparameter space.

Chapter 3: Methodology

The purpose of this chapter is to introduce the methods used in this thesis. First, conduct data collecting and cleaning to isolate out the relevant factors for aircraft delays. Second, build train and test models for machine learning-based classification problems. Third, adjust the features then focus on which method to predict the aircraft delays with the highest accuracy.

3.1 Data Collection and Cleaning

Provided by Kaggle, the data consists of multiple files of airline, weather, airport, and employment information. The data that were collected are weather, passenger, aircraft, coords (coordinate for airports), names of carrier, and employee. The weather data is provided by the National Centers for Environmental Information (NOAA), while, the flight data is provided by the Bureau of Transportation statistics (BTS, 2023). The data are merged together using date, time, and the link to all of the information. For duplicate data in airport coordinates, names of the carrier, aircraft tail number, and origin airport ID, they will be dropped.

There are a total of 77,350 lines for weather data. The weather information includes station name, date, average wind speed, peak gust time, precipitation, average temperature, maximum temperature, minimum temperature, fog, heavy fog, thunder, ice pellets, hail, glaze, dust, smoke, blowing or drifting snow, tornado, high winds, blowing spray, mist, drizzle, freezing drizzle, rain, freezing rain, snow, unknown precipitation, ground fog, and ice fog. Most of the weather information is very specific and not a number or NaN. Every time there is an occurrence of a specific weather phenomenon, it is recorded as 1. Therefore, with the current set of data, NaN is replaced with 0.

For general missing weather data such as minimum temperature, average temperature, and average daily wind speed, the data is replaced with median for temperature related data and mean. The weather data is provided as a daily summary at the local reporting station. Therefore, as an improvement in both departure and arrival, the weather data will be utilized for both.

Next, we define a function. It performs feature engineering, data merges, and cleanup, using one month of on-time data at a time, which is from the Bureau of Transportation Services. The parameters include monthly_data, aircraft, airport coordinates, names, weather, passengers, and employees. The function

also returns cleaned data of one month of on-time reporting. This step includes starting a timer to track how long the cleaning function takes, cleaning up by dropping rows with no departure time, tail number, or canceled. Then, we list flight segment numbers for daily flight segments by tracking tail number, listing the number of concurrent flights at the airport in the time block, and combining the number of seats with the main frame on tail number. The missing aircraft will be filled with the average.

After that, we change the data type for the number of seats to reduce the memory usage, merge to get the proper carrier's name, add monthly flight statistics for carrier, airport information, and airport flight per month, and add monthly passenger statistics for carriers and airports. Then, we merge the employee, flight attendants per passenger, and ground service per passenger to the data set.

For the plane age data, it is calculated by subtracting the current year with the manufacture year of the aircraft (the current year - the manufacture year of the aircraft). The airport coordinates of latitude and longitude of the departing airport are merged and added to the data. The airport that has flight traffic of less than 10th percentile is dropped due to lacking the amount of traffic when compared to other airports. Meaning that airports that have less than 1,100 flights per month are dropped as these airports are capable of handling manageable traffic volume. For the selected airports, the weather data is added. After that, the flight data is merged and the columns that are not used are dropped. Table 3 below has shown different features of the data collected and their types.

Table 3 Features and types of data

Feature	Data Type	Description
Station name	String	Name of the weather station reporting the data.
Date	String	Date of the weather report.
Average wind speed	Float	Average wind speed recorded.
Peak gust time	String	Time of the peak wind gust.
Precipitation	Float	Amount of precipitation.
Average temperature	Float	Average temperature recorded.
Maximum temperature	Float	Maximum temperature recorded.
Minimum temperature	Float	Minimum temperature recorded.
Fog	Integer	Presence of fog (1 if present, 0 if absent).
Heavy fog	Integer	Presence of heavy fog (1 if present, 0 if absent).
Thunder	Integer	Presence of thunder (1 if present, 0 if absent).
Ice pellets	Integer	Presence of ice pellets (1 if present, 0 if absent).
Hail	Integer	Presence of hail (1 if present, 0 if absent).
Glaze	Integer	Presence of glaze (1 if present, 0 if absent).
Dust	Integer	Presence of dust (1 if present, 0 if absent).
Smoke	Integer	Presence of smoke (1 if present, 0 if absent).
Blowing or drifting snow	Integer	Presence of blowing or drifting snow (1 if present, 0 if absent).
Tornado	Integer	Presence of tornado (1 if present, 0 if absent).
High winds	Integer	Presence of high winds (1 if present, 0 if absent).
Blowing spray	Integer	Presence of blowing spray (1 if present, 0 if absent).
Mist	Integer	Presence of mist (1 if present, 0 if absent).
Drizzle	Integer	Presence of drizzle (1 if present, 0 if absent).
Freezing drizzle	Integer	Presence of freezing drizzle (1 if present, 0 if absent).
Rain	Integer	Presence of rain (1 if present, 0 if absent).
Freezing rain	Integer	Presence of freezing rain (1 if present, 0 if absent).
Snow	Integer	Presence of snow (1 if present, 0 if absent).
Unknown precipitation	Integer	Presence of unknown precipitation (1 if present, 0 if absent).
Ground fog	Integer	Presence of ground fog (1 if present, 0 if absent).
Ice fog	Integer	Presence of ice fog (1 if present, 0 if absent).
Flight Data (Various features)	Mixed	Features such as departure time, tail number, flight segment numbers, concurrent flights, number of seats, flight statistics, passenger statistics, employee data, plane age, airport coordinates, and weather data. The data types range from strings to integers of different bit sizes (e.g., 'int8', 'int64', 'int32') based on the specific feature and its representation.

Table 4 Different features of collected data

Airport	Aircraft	Carrier	Ontime Reporting	Carrier	Weather
Airport ID	Manufacture year	Airline ID	Month	Year	Precipitation
Airport Name	Tail number	Carrier Name	Day	Airline ID	Snowfall
Origin City		Service Class	Week	Carrier Name	Snow on ground
Name		No. Departures perform per year		Pilot/Copilot	Max temperature
		Passengers enplaned for year		Maintenance	Max Wind speed for day
				Aircraft Control	Daily total sunshine
				Passenger Handling	Fastest 5 seconds wind speed
				Trainee and Instructor	Fog
				Traffic Solicitors	Heavy Fog
				Transport Related	Thunder
				Others employee	Ice Pellets
				General Manager	Hail
				Other flight personnel	Smoke Haze
				General Services & Administration	Tornado

Next, the data types for all the various fields are cleaned up to reduce memory usage. Month and day of the week are defined as ‘object’. For delay, distance group, and segment numbers are defined as ‘int8’. Airport flights per month, airline flight month and airline airport flight per month are defined as ‘int64’.

Lastly, the plane age is defined as type ‘int32’. The timer is stopped and elapsed time is calculated. The train and test sets are generated. The train set is split into subsets and the validation set is generated.

Specific weather information for each arrival and departure location is assigned for the accuracy of the prediction because the weather at both arrival and departure will affect whether the aircraft is going to be delayed.

After combining all the data into one file, there are some data columns that are text and not numbers. The method chosen to transform these data is OrdinalEncoder although there are also OnehotEncoder and LabelEncoder. The OrdinalEncoder assigned an integer value to a category. It is easily reversible. It will assign integers to labels in the order that it observes in the data. OnehotEncoder creates a binary column for each category and returns an array of numbers. The LabelEncoder transforms the data from text providing values between 0 and n - 1. The OrdinalEncoder will find the unique values per feature and transform the data. (learn)

Table 4 in the preceding of this thesis offers a comprehensive summary of various weather variables, encompassing a diverse range of meteorological elements crucial in understanding climatic conditions. Building upon this foundational knowledge, Table 5 aims to further elucidate and define these weather variables, providing a detailed breakdown of their specific characteristics and measurements. In this section, the definitions of each weather variable outlined in Table 3 will be expanded upon, offering a more in-depth exploration of their individual attributes. This comprehensive elaboration seeks to enhance the understanding of these meteorological factors, laying a robust groundwork for their utilization and interpretation in subsequent analyses or research within the realm of climatic studies or related fields.

Table 5 Definition of different weather variables

Abbreviation	Description
PRCP	Precipitation
SNOW	Snowfall
SNWD	Snow depth
TMAX	Maximum temperature
TMIN	Minimum temperature
AWND	Average daily wind speed
FMTM	Time of fastest mile or fastest 1-minute wind
FRGB	Base of frozen ground layer
FRGT	Top of frozen ground layer
FRTH	Thickness of frozen ground layer
PGTM	Peak gust time
PSUN	Daily percent of possible sunshine
SN*#	Minimum soil temperature
SX*#	Maximum soil temperature
THIC	Thickness of ice on water
TOBS	Temperature at the time of observation
TSUN	Daily total sunshine
WDF1	Direction of fastest 1-minute wind
WDF2	Direction of fastest 2-minute wind
WDF5	Direction of fastest 5-second wind
WDFG	Direction of peak wind gust
WDFI	Direction of highest instantaneous wind
WDFM	Fastest mile wind direction
WDMV	24-hour wind movement
WESD	Water equivalent of snow on the ground
WESF	Water equivalent of snowfall
WSF1	Fastest 1-minute wind speed
WSF2	Fastest 2-minute wind speed
WSF5	Fastest 5-second wind speed
WSFG	Peak gust wind speed
WSFI	Highest instantaneous wind speed
WSFM	Fastest mile wind speed
WT**	Weather Type
WVxx	Weather in the Vicinity

3.2 Research Analysis

Next is choosing what information is important and what should be eliminated. At first, the data are combined together into one big data. In terms of the data cleaning, the data are split into training, cross validation, and test datasets. For irrelevant information such as previous airport, departing airport, carrier name, and departure time, they are excluded from the training data. All of the data that are included in the training data are quantitative. The data for the training portion are included as much as possible as long as they are within the computing power of the machine.

Use the function ".describe()" to find each of the data columns and put into visualization based on statistical data of count, mean, standard deviation, min, 25%, 50%, 75%, max, skewness, and also kurtosis. The data are then verified once again that there is no NaN and the data that have an impact on the flight delay are selected. The data include month, day of the week, departure time, distance group, segment number, concurrent flight, number of seats, carrier name, airport flight per month, airline flight per month, airline airport flight per month, average monthly passenger at specific airport, flight attendants per passenger, ground service per passenger, the aircraft age, departing airport, latitude, and longitude of the airport.

For the weather information, there are precipitation, inch of snowfall for day, inch of snow on ground for day, maximum wind speed for day, maximum temperature for day, lowest temperature for day, average temperature for day, direction of fastest 2-minute wind (Degree), direction of fastest 5-second wind (Degree), fastest 2-minute wind speed (tenths of meters per second), and fastest 5-minute wind speed (tenths of meters per second). All of the data are included in the generating training and test datasets. The aircraft that have arrived more than 15 minutes late is then highlighted for further examination.

Figure 15 has shown the heat map that is used to map out the relevant information from the data to include and exclude what is necessary in order to make a non-biased model out of the data. The chosen data is then used to make the train data set.

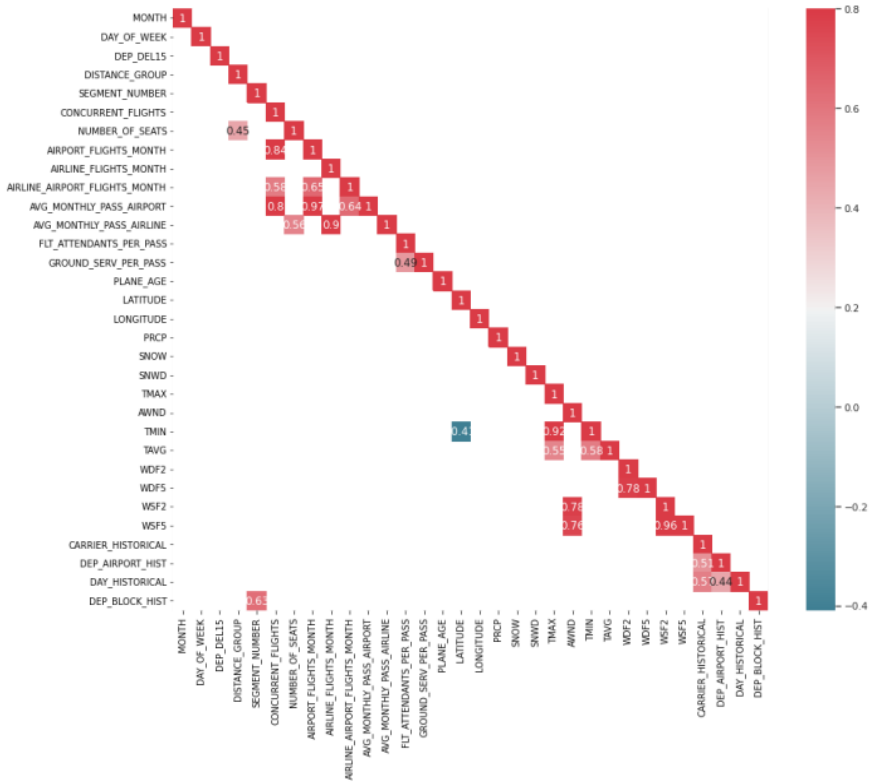


Figure 15: Heat map

3.3 Findings

The amount of the data is large, but the amount of the data specifically for weather is limited. It is crucial not to be biased toward the information that is collected. Therefore, most of the selection is done by the program. Each of the data in the column does not have an even distribution. Standardscaler is used in order to standardize the feature by utilizing the mean and scaling to unit variance (learn). The equation for standard scaler is in Equation (4).

$$z = \frac{(x-\mu)}{\sigma} \quad (4)$$

where z = Z-Score

x = Given data value

μ = Mean

σ = Standard deviation

After transforming and standardizing all of the data, the next step is to select the data for training and testing after the initial analysis from reducing the non-contributing variables. Flight delays other than weather may be excluded. It is determined by how relevant it is to the contribution of flight delays. When determining which algorithm to use for this dataset, the obtain data set has a labeled data, structure, and unprocessed. Supervised machine learning algorithm is the preferred method.

There are 74 features total and only 3 features were not selected. The 3 features are month, day of the month, and day of the week. The number of total delay data as compared to all of the data were plotted in order to see if the ratio is still the same from the original data. Once the data has been verified, the training can begin making the final model. The result is shown in Figure 16.

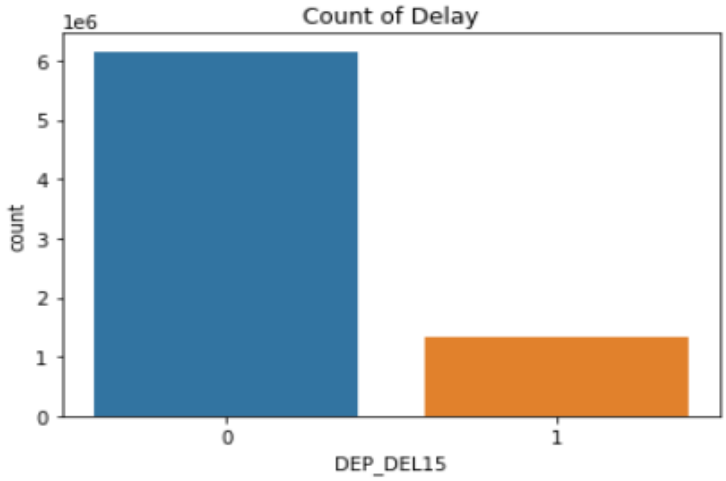


Figure 16: Number of Train Data of No Delay (0) and Delay (1)

To figure out the important features of the data, a Random Forest feature is used in this case, which can be seen in Figure 17. The output of this feature provides the relative importance or the contribution of each feature in the prediction. It is important to see what is the importance of data that was calculated in the model. Feature importance in a Random Forest model serves as a metric to assess the relative significance of different input variables in influencing the model's predictions. It functions by analyzing the contribution of each feature to the predictive power of the ensemble of decision trees. This determination is based on how effectively a feature reduces uncertainty or impurity in the nodes when splitting the data across numerous trees. The algorithm calculates the importance of each feature by examining the average decrease in impurity for that feature across all the trees in the forest. By

quantifying the impact of variables on the model's accuracy, feature importance aids in identifying the most influential factors driving predictions. This analysis assists in prioritizing important features, guiding feature selection processes, and providing insights into the underlying relationships between variables and predicted outcomes. It is crucial to note that while feature importance highlights the relevance of variables in the model, it does not establish causation between these features and predictions, serving as a tool for model interpretation and refinement. Understanding the importance of each feature provides valuable insights into the inner workings of the model, aiding in model interpretability and feature selection. When some features show low importance, it suggests potential opportunities for model optimization through dimensionality reduction or feature engineering. It can also guide decision-makers and domain experts by highlighting the most influential factors in predictive modeling. The results of feature importance analysis will provide the next suggested action, whether it is refining the model, or enhancing feature for further studies. In essence, Random Forest feature importance provided the tools to extract meaningful knowledge from the models and use that knowledge to drive better outcomes and informed choices in various fields and applications.

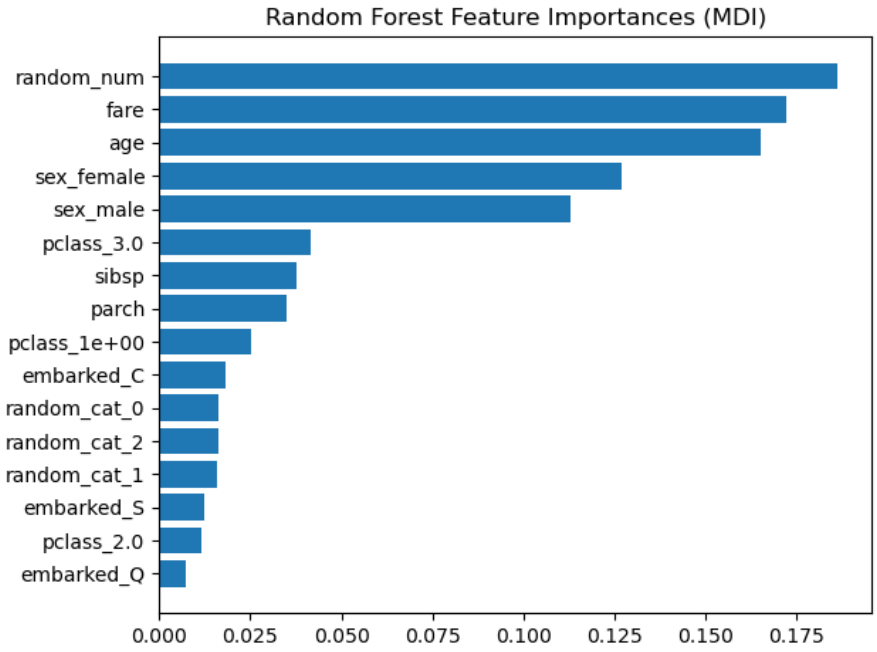


Figure 17: Random Forest Feature Importance Sample Result (Medium, 2020)

The Shapley (SHAP) values offer a deeper and more comprehension of feature importance. An example of SHAP values can be seen in Figure 18.

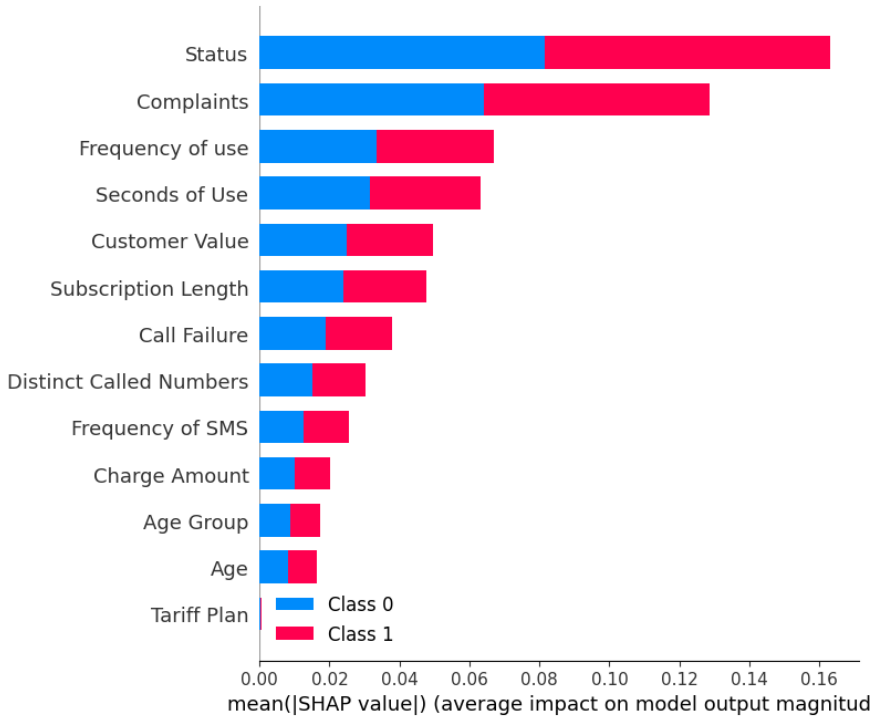


Figure 18: SHAP Values Sample Result (Datacamp, 2022)

While traditional feature importance provides an initial understanding of which features are influential in a model's predictions, SHAP values take this analysis to a granular level. By going into the specific contribution of each feature to individual predictions, considering interactions among features. This level of detail is crucial as it uncovers not just the importance of features but also how each feature affects predictions in different scenarios. SHAP values offer insights into the direction and magnitude of a feature's impact, aiding in understanding complex relationships within the model. By highlighting the influence of each feature on individual predictions, SHAP values assist in model debugging, validation, and refinement. They also aid in feature selection by identifying truly impactful features while uncovering potential redundancies or confounding effects. This comprehensive understanding provided by SHAP values enhances the interpretability and trustworthiness of Random Forest models, allowing for more informed decisions in model improvement and feature engineering. Ultimately, SHAP values play a pivotal role in not just gauging feature importance but in empowering users to enhance the model's performance and transparency.

The hyperparameters to be tuned of Random Forest classifier are n_estimators (list of tree classifiers), Max depths, Max features, Min_samples_leaf, Min_sample_split and Criterion. Tuning hyperparameters with GridSearchCV is to find the best settings for a machine learning model. It's beneficial because it does this in a smart and efficient way. Instead of trying random settings, it checks all possible combinations. This makes the model work better and avoids problems like making it too good at the training data but not so good with new data. GridSearchCV also keeps things organized and saves time, making it easier for people who work with machine learning to make their models perform at their best. These hyperparameters are can be shown in Table 6.

Table 6 Random Forest Classifier Hyperparameters and Values

Hyperparameter	Value
N_estimators	150, 200, 300
Max depths	12, 15, 20, 30, 40, NaN
Max features	Sqrt, Auto
Min_samples_leaf	3,5,10,20
Min_sample_split	2,5,10,20,30
Criterion	Gini, Entropy

GridSearchCV was performed to find the combination of the best hyperparameters. The most conservative settings were employed for the hyperparameters to ensure that no shortcuts were taken in the calculations or to compromise the results. The other parameters used are as follows: min_weight_fraction_leaf = 0, max_leaf_nodes = None, min_impurity_decrease = 0, bootstrap = True, oob_score = False, n_jobs = None, random_state = None, verbose = 0, warm_start = False, class_weight = None, ccp_alpha = 0, max_samples = None.

The hyperparameters to be tuned of AdaBoost classifier were n_estimators, learning rate, Algorithm, base_estimator_criterion and base_estimator_splitter. These hyperparameter and values can be shown in Table 7.

Table 7 AdaBoost Classifier Hyperparameters and Values

Hyperparameter	Value
N_estimators	100, 200, 300
Algorithm	SAMME, SAMME.R.
Learning Rate	0.001, 0.01, 0.1,0.25, 1
Base_estimator_criterion	Gini, entropy
Base_estimator_splitter	Best, random

The hyperparameters to be tuned of CatBoost classifier were depth, the learning rate and iterations. These hyperparameters and values can be shown in Table 8.

Table 8 CatBoost Classifier Hyperparameters and Values

Hyperparameter	Value
Depth	6, 8, 9, 10, 15, None
Iterations	100, 300, 500, 700, 900
Learning Rate	0.01, 0.1, 0.5, 1

The hyperparameters to be tuned of Gradient Boosting classifier were n_estimators, learning rate, min_sample_leaf, min_sample_split, max_depth, max_features, criterion, and subsample. These hyperparameters and values can be shown in Table 9.

Table 9 Gradient Boosting Classifier Hyperparameters and Values

Hyperparameter	Value
N estimator	10, 50, 100, 300, 500
Learning Rate	0.01, 0.025, 0.05, 0.1, 0.5
Min sample leaf	1, 5, 12
Min sample split	1, 5, 12
Max_depth	3, 5, 10, 15, None
Max_features	Log2, sqrt
Criterion	Friedman_mse, squared_error
Subsample	0.5, 0.85, 0.95, 1

3.4 Evaluation Metrics

To provide a standardized way to measure and assess the performance of the model, we rely on essential evaluation metrics. The sample results of the prediction model are presented in the classification report below, which includes crucial insights into the model's performance.

The key components to provide a standardized way to measure and assess the performance of the model are these evaluation metrics. The sample results of the prediction model are shown in the classification report below.

	precision	recall	f1-score	support
0	0.93	0.99	0.96	737142
1	0.94	0.67	0.78	171321
accuracy			0.93	908463
macro avg	0.93	0.83	0.87	908463
weighted avg	0.93	0.93	0.93	908463

Figure 19: Sample of Classification Report

For this classification report, the results are shown in a confusion matrix. The confusion matrix provides the prediction results in Table 10.

Table 10 Confusion Matrix

		Actual Values	
		1	0
Predicted Values	1	TP	FP
	0	FN	TN

Key Definitions:

TP (True Positive): The model predicts delays, and the collected data confirms that the flight is delayed.

FP (False Positive): The model predicts delays, but the data shows no delay.

TN (True Negative): The model predicts no delays, and the data confirms that the flight is not delayed.

FN (False Negative): The model predicts no delays, but the data confirms a delay.

As shown in the confusion matrix above, actual and predicted values show the precision and recall accuracy of the data. True positive in this case is that the model predicts delays and the data collected says that the flight is delayed. For true negative, on the other hand, the model predicts no delay and the data collected says the flight is not delayed. False positive is when the model predicts delays, but the data is not delayed. Lastly, false negative is when the model predicts no delay, but the data is delayed.

For a good model, the true positive and true negative should have a higher number, while false negative and false positive should have a lower number. Although having higher false positive provides higher or over readiness to the system, a false negative is a miss for the model.

Precision is the ratio of true positive over predicted positives including true positive and false positive. Precision shows how accurate the model is. Lower precision identifies the model detecting false positive.

$$Precision = \frac{TP}{TP+FP} \tag{5}$$

Recall is the ratio of true positive over actual positive including true positive and false negative. Recall is how sensitive the model to false negative. The lower the value means the lower quality of the model. The lower value represents the model with high numbers of false negative.

$$Recall = \frac{TP}{TP+FN} \tag{6}$$

Accuracy is the ratio of correct prediction over all of the predictions, both true and false prediction. F1-score, which is the combination of precision and recall, is a better presentation especially the macro average of the F1-score. The macro average takes all of the F1-score into account. Although there is also a weighted average on the table to be fair, the data should be treated equally, so the same weight should be considered throughout. As a result, the macro average is the preferred method in comparing between each of the data set. If there is a very small difference in the F1-score then recall will be use as a tie breaker to determine which model is the best. In Equation 7, the equation of recall signifies the amount of false negative.

$$Accuracy = \frac{TP+TN}{TP+FN+TN+FP} \tag{7}$$

F1-score: The F1-score combines precision and recall to provide a single value that balances both aspects. It is particularly useful when comparing models. The macro average of the F1-score takes all F1-scores into account equally.

$$F1 - score = \frac{2\ x\ Precision\ x\ Recall}{Precision + Recall} \quad (8)$$

A higher value of recall means lower false negative in the model. A false negative in this case is when a model predicts that the plane is not going to be delayed, but it is delayed. This is the worst prediction because it could cause further unexpected delays throughout the whole day. On the other hand, true negative is an over estimation which could in turn waste the extra buffer time in the schedule, but will mostly prevent propagated delay for the rest of the schedule.

Chapter 4: Results and Discussion

The purpose of this chapter was to present results and a brief explanation from GridSearchCV for each algorithm. In this comparison, we evaluated four different methods for flight delay prediction: Random Forest, AdaBoost, CatBoost, and Gradient Boosting. These methods employed different combinations of hyperparameters, which were selected using GridSearchCV, and produced varying accuracy results.

Random Forest was an ensemble learning method that combined multiple decision trees to make predictions. In this case, Random Forest used 150 estimators, representing the number of decision trees in the ensemble. The max_depth parameter was set to NaN, allowing the trees to grow until all leaves were pure or the minimum samples split criterion was met. The max features parameter was set to "sqrt," meaning the square root of the total number of features was considered for splitting at each node. The minimum samples per leaf and minimum samples per split were set to 3 and 10, respectively. The selected criterion was Gini. The Random Forest model achieved an accuracy of 83 percent.

The process of adjusting hyperparameters using GridSearchCV within the Random Forest method unveiled an intriguing trend, particularly when focusing on optimizing 'n_estimators.' The observed behavior revealed a rather unexpected outcome: as 'n_estimators' increased from 150 to 200 and further to 300, the corresponding accuracy scores exhibited a decrement. Specifically, the accuracy values for these increments were 0.81292, 0.81288, and 0.81258, respectively. This pattern indicates a critical insight – contrary to the anticipated notion that higher 'n_estimators' would universally enhance accuracy, there exists a point of diminishing returns in this context. Simply utilizing the maximum 'n_estimators' across all Random Forest models might not represent the most effective solution. This observation underscores the need for a nuanced approach to hyperparameter optimization, suggesting that an excessively high number of trees could lead to a reduction in model efficacy rather than improvement. Hence, a strategic selection of 'n_estimators,' possibly opting for a lower value within the tested range, could potentially yield a more optimal model performance, balancing complexity and accuracy more effectively. This outcome emphasizes the importance of meticulous hyperparameter tuning and the consideration of interactions between parameters to derive the most efficient and accurate model for the given dataset.

Gradient Boosting was an algorithm that combined multiple weak learners to create a strong predictive model. In this case, Gradient Boosting used 500 estimators, the highest number among all the algorithms considered. The learning rate for Gradient Boosting was set to 0.05, striking a balance between convergence speed and accuracy. The min sample leaf was 1, and the min sample split was 1 as well. The subsample was also 1. The algorithm used the friedman_mse criterion for splitting. The max depth was set to 3, and the max features parameter was set to "sqrt." The Gradient Boosting model achieved an accuracy of 81.1 percent.

CatBoost was a Gradient Boosting algorithm designed to handle categorical features effectively. Like AdaBoost, CatBoost used 300 iterations. The learning rate for CatBoost was set to 0.01, which was significantly lower than the other algorithms. CatBoost depth was 10, which was the maximum. The CatBoost model achieved an accuracy of 81 percent.

AdaBoost was an adaptive boosting algorithm that trained weak learners and combined their predictions to create a strong ensemble. In this case, AdaBoost used 300 estimators, twice the number used in Random Forest. The learning rate was set to 0.25, which determined the contribution of each weak learner in the final prediction. The estimator criterion used was entropy, and it was employed to select the best estimator splitter. The best algorithm was SAMME. The Base estimator splitter that was chosen was best. AdaBoost did not specify max depth or max features. The accuracy achieved by the AdaBoost model was 72.6 percent. The comparison of each algorithm can be found in Table 11.

Table 11 Algorithm Comparison

Random Forest	GradientBoost	CatBoost	AdaBoost
n_estimator: 150	n_estimator: 500	n_estimator: 300	n_estimator: 300
Max_depth: NaN	learning rate: 0.05	learning rate: 0.001	learning rate: 0.25
Max_features: sqrt	friedman_mse criterion	Algorithm: SAMME.R	Estimator_criterion: entropy
Min_samples_leaf: 3	Max_depth: 3	-	Best_estimator_splitter: best
Min_samples_split: 10	Max_features: sqrt	-	-
Accuracy: 83%	Accuracy: 81.1%	Accuracy: 81%	Accuracy: 72.6%
F1-Score: 0.56	F1-Score: 0.47	F1-Score: 0.46	F1-Score: 0.45

Among the four algorithms, Random Forest had achieved the highest accuracy of 83%, followed by Gradient Boosting at 81.1%, CatBoost at 81%, and AdaBoost at 72.6%. These results indicated that Random Forest had

outperformed the other algorithms in terms of accuracy. However, it was important to consider other factors such as model complexity, computational resources required, and interpretability when choosing the best algorithm for a specific task.

Random Forest and Gradient Boosting had utilized the number of estimators in their provided range, with 150 and 500, respectively, suggesting a stronger ensemble with increased model capacity. AdaBoost and CatBoost had employed 300 estimators, providing a balance between accuracy and computational efficiency.

Regarding the learning rate, Random Forest did not explicitly specify a learning rate, while AdaBoost had employed a learning rate of 0.25. CatBoost had used a significantly lower learning rate of 0.001, and Gradient Boosting had set the learning rate at 0.05. The choice of learning rate was crucial, as it affected the convergence speed and model performance. Fine-tuning the learning rate for each algorithm could help optimize their performance.

In terms of feature selection, Random Forest and Gradient Boosting had used "sqrt" as the max features parameter, indicating that the square root of the total number of features was considered for splitting at each node. CatBoost did not have the specific max feature parameters, and AdaBoost did not have any max features. Experimenting with different max feature settings might provide further insights into model performance and diversity.

When comparing the F1-score for AdaBoost under various conditions, such as precision and recall, it consistently demonstrated lower values compared to the Random Forest method. This outcome indicated that the Random Forest served as the established baseline method for adjusting different features across diverse tests. For a comprehensive comparison, the other two methods need evaluation against the current best-performing results. They would be compared against the Random Forest model's predictions, assessing accuracy and F1-score. A thoroughly examined forest model generally yields superior predictions than employing a single stump for constructing the model. It’s evident that there is room for enhancing and fine-tuning the model. Adjusting model features is crucial for improving accuracy, though there's a point where excessive feature adjustments might compromise accuracy. Figures 20 and 21 illustrate the results from the Random Forest and AdaBoost classification

models, respectively. Following these, Figures 22 and 23 showcase the outcomes of CatBoost and Gradient Boosting, respectively.

	precision	recall	f1-score	support
0	0.83	0.99	0.9	182210
1	0.72	0.13	0.22	42790
accuracy			0.83	225000
macro avg	0.77	0.56	0.56	225000
weighted avg	0.81	0.83	0.77	225000

Figure 20: Confusion Matrix of Random Forest

	precision	recall	f1-score	support
0	0.81	1.00	0.90	182210
1	0.53	0.01	0.01	42790
Accuracy			0.81	225000
macro avg	0.67	0.50	0.45	225000
weighted avg	0.76	0.81	0.73	225000

Figure 21: Confusion Matrix of AdaBoost

	precision	recall	f1-score	support
0	0.77	0.86	0.81	182210
1	0.84	0.75	0.02	42790
accuracy			0.80	225000
macro avg	0.69	0.50	0.46	225000
weighted avg	0.76	0.81	0.73	225000

Figure 22: Confusion Matrix of CatBoost

	precision	recall	f1-score	support
0	0.81	1.00	0.90	182210
1	0.65	0.02	0.04	42790
accuracy			0.81	225000
macro avg	0.73	0.51	0.47	225000
weighted avg	0.78	0.81	0.73	225000

Figure 23: Confusion Matrix of Gradient Boosting

The evaluation of machine learning algorithms, Random Forest, AdaBoost, CatBoost, and Gradient Boosting, revealed distinctive performance metrics. Random Forest demonstrated an impressive accuracy, with an F1-score of 0.56, indicating a balanced measure of precision and recall. Its precision of 0.77 highlights its ability to make accurate positive predictions, while a recall of 0.56 indicates it effectively captures true positives. Gradient Boosting achieved an F1-score of 0.47, boasting a precision of 0.73 and a recall of 0.51. CatBoost closely followed with an F1-score of 0.46, a precision of 0.69, and a recall of 0.50. Finally, AdaBoost, on the other hand, yielded a slightly lower F1-score of 0.45, with a precision of 0.67 and recall of 0.50. These metrics collectively provide a comprehensive view of each algorithm's ability to balance precision and recall in the context of aircraft delay prediction, enabling data scientists and practitioners to make informed choices regarding model selection and optimization.

To further understand the error analysis of the result of Random Forest report, the False Positives emerge as outliers that deviate markedly from the mean or typical values within the dataset. These anomalies are characterized by their placement at the far ends of the spectrum in terms of feature values, specifically within the parameters of historical departure block, time block of departure, airport history, distance, and carrier data. This peculiarity signifies instances where the model inaccurately predicted a flight as "Not Delayed" when it was, in fact, delayed. These outliers suggest a unique behavior within the dataset, indicating the need to address these exceptional cases to refine the model's accuracy. Addressing the disparities within these influential features could potentially mitigate misclassifications and enhance the model's predictive capabilities.

The consistency between the F1-score and accuracy results further validates the model rankings, demonstrating a robust performance across multiple evaluation metrics. With Random Forest leading in both accuracy and F1-score at 0.56,

followed by GradientBoost at 0.47, CatBoost at 0.46, and Adaboost at 0.45, the alignment reinforces the models' relative effectiveness in predictive performance. Using F1-score as an additional evaluation metric provides an alternate perspective, verifying the models' robustness beyond just accuracy. F1-score's emphasis on striking a balance between precision and recall offers insight into the models' abilities to manage false positives and false negatives. This convergence between accuracy and F1-score underscores the reliability of the Random Forest model and the overall consistency of model rankings, validating their performance across different evaluation perspectives and affirming their effectiveness in handling various data intricacies and variables within the dataset.

Transitioning to Table 12, the Random Forest Confusion Matrix offers a detailed breakdown of the model's classifications, unveiling insights beyond summary metrics. This comprehensive view delineates true positives, true negatives, false positives, and false negatives, enriching our understanding of the model's precision and recall. It highlights the balance between accurately identified instances and misclassifications within the model's predictions.

Table 12 Randon Forest Confusion Matrix

		Actual Value	
		1	0
Predicted Values	1	5646	37144
	0	2221	179989

The analysis of the confusion matrix would be as follows:

True Positives (TP): The model accurately predicted "Delayed" (Class 1) in 5,646 instances.

False Positives (FP): Instances where the model incorrectly predicted "Delayed" (Class 1) when the actual class was "Not Delayed" (Class 0) amount to 37,144 cases.

False Negatives (FN): The model incorrectly predicted "Not Delayed" (Class 0) when the actual class was "Delayed" (Class 1), constituting 2,221 cases.

True Negatives (TN): Instances where the model correctly predicted "Not Delayed" (Class 0) amount to 179,989 cases.

The obtained F1-score of 0.65 and an accuracy of 0.81 from testing a different dataset using the same Random Forest classification model showcases a relatively robust performance in predicting aircraft delays. This indicates that the model has retained its predictive power when applied to a new dataset, maintaining a reasonably balanced trade-off between precision and recall. Considering the F1-score being slightly lower than the accuracy, it suggests the model might have a proportionally higher number of false negatives (missed delayed flights) than false positives (non-delayed flights incorrectly labeled as delayed). As for the ratio of aircraft not delayed to delayed aircraft in the new dataset, it's reasonable to estimate it to be around 4:1 or 5:1, considering the F1-score and accuracy values and the model's typical performance in classifying imbalanced datasets.

Feature Importance

Random Forest calculates feature importance by assessing how much each feature contributes to the reduction of impurity within the decision trees that make up the ensemble. This importance score is based on factors such as the number of times a feature is used to split nodes and the decrease in impurity achieved when using a specific feature.

Random Forest feature importance helps pinpoint which variables in the dataset have the most influence on predicting aircraft delays. By examining the importance scores, such as departure time, temperature, or precipitation, have the greatest impact on the model's predictions. This information can be instrumental in decision-making and operational efficiency within the aviation industry, ultimately reducing delays and improving the traveler experience.

The analysis of the first 51 features from the Random Forest feature importance on aircraft arrival delay provided valuable insights into the factors that significantly influenced the timeliness of aircraft arrivals time. These features were ranked in order of importance and offered a comprehensive understanding of the various variables that contributed to arrival delays, as shown in Figure 24.

The process of removing lower-ranking features from a Random Forest model often presents nuanced outcomes, typically resulting in either minimal impact on model performance or marginal improvements in computational efficiency. Features like WT05 (Hail), WT09 (Blowing or Drifting Snow), WT04 (Ice pellets), WT02 (Heavy fog), Snow_Dest, WT06 (Glaze or rime), WT01 (fog), while ranking lower in importance according to the model, carry specific meteorological details that might contribute to the accuracy or F1-score in particular instances. The removal of these seemingly less influential features

may not significantly affect the model's immediate predictive capability or computational efficiency. However, it is crucial to acknowledge the potential loss of distinctive and contextually relevant information embedded within these variables. They could encompass rare or critical meteorological occurrences that, although infrequent, might significantly impact the model's ability to generalize or predict specific events accurately. Therefore, while the immediate impact of their removal might seem minor, it is essential to carefully weigh the potential loss of unique details against any computational efficiency gained. Thorough post-removal analysis and validation become imperative to gauge the real impact on model performance, ensuring that the removal of these lower-ranking features does not compromise the model's ability to capture critical weather-related nuances that could influence its predictive power in certain scenarios.

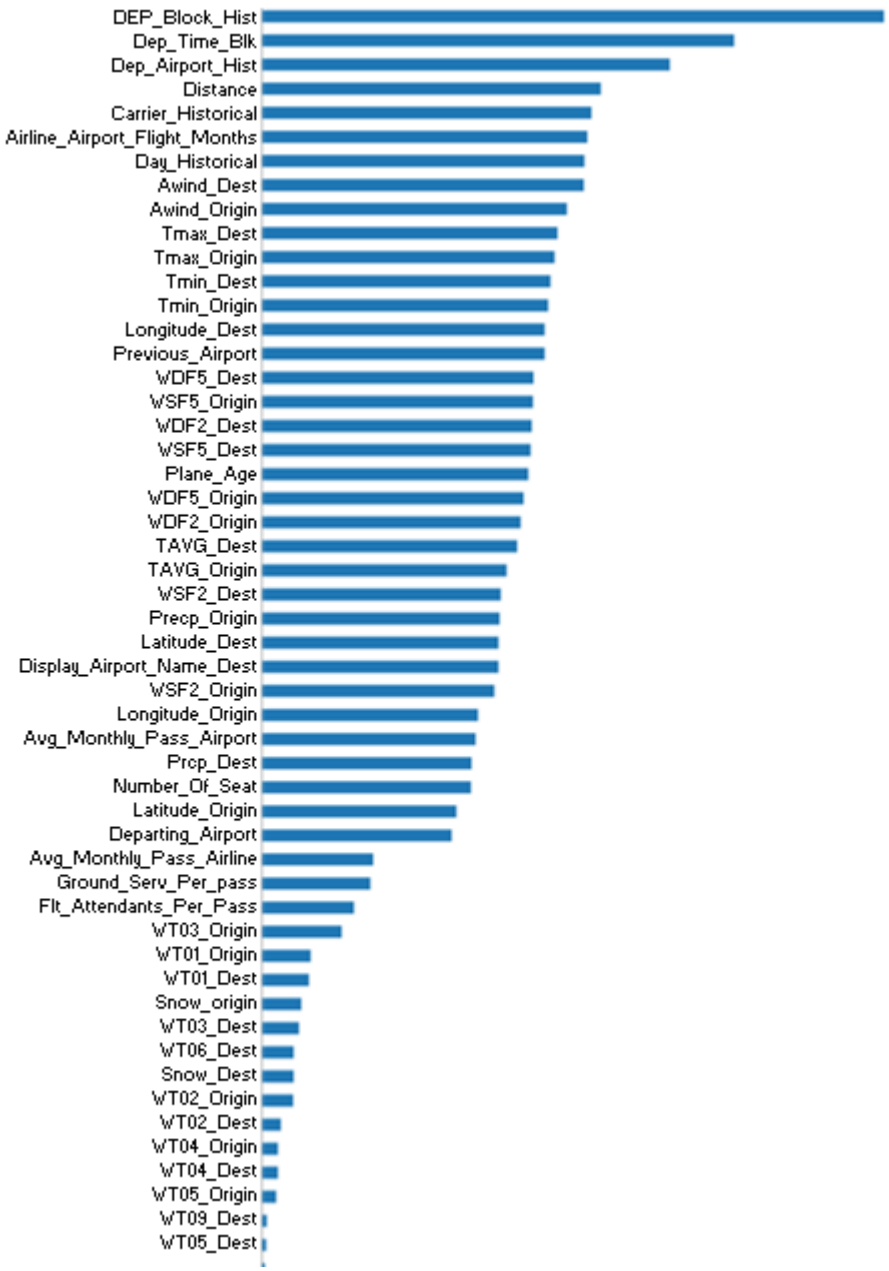


Figure 24: Random Forest Feature Importance

One of the most important features was DEP_Block_Hist, which represented the historical departure block time. Deviations from the scheduled departure block time could impact arrival delays, highlighting the importance of adhering to departure schedules. Dep_Time_Blk, another crucial feature, categorized the actual departure time into specific time intervals. Departure time could influence arrival delays, as delays in departures might have had a cascading effect on subsequent flight operations.

Dep_Airport_Hist, on the other hand, represented the historical performance of the departure airport. Factors such as congestion, efficiency, and operational practices at the departure airport could contribute to arrival delays. Similarly, Distance played a vital role in arrival delays, as longer flight distances might have required more time for travel, increasing the likelihood of delays.

Carrier_Historical was an essential feature that reflected the historical performance of the airline carrier operating the flight. Factors such as operational efficiency, on-time performance, and maintenance practices of the carrier could impact arrival delays. Moreover, Airline_Airport_Flight_Months indicated the number of flights the airline had been operating at a specific airport. This feature suggested that airlines with more experience operating at an airport might have had better operational strategies, potentially reducing arrival delays.

Day_Historical considered the historical performance of the day of the week. Certain days might have experienced higher passenger volumes or air traffic, leading to potential delays. Additionally, factors other than weather played a significant role in arrival delays. These included features like Awind_Dest and Awind_Origin, which represented the average wind speed at the destination and origin airports, respectively. Wind conditions could affect aircraft performance, including takeoff, landing, and overall flight time, influencing arrival delays.

Temperature variables such as Tmax_Dest, Tmax_Origin, Tmin_Dest, and Tmin_Origin indicated the maximum and minimum temperatures at the destination and origin airports. Extreme temperatures could impact aircraft operations, potentially causing delays. Similarly, factors other than weather, such as precipitation, represented by Precp_Origin, at the origin airport could also affect flight operations. Adverse conditions like rain or snow could impact arrival delays.

Longitude and Latitude coordinates of the destination and origin airports (Longitude_Dest, Longitude_Origin, Latitude_Dest, Latitude_Origin) had an impact on air traffic patterns and weather conditions. These geographical factors could influence arrival delays. Moreover, features like WSF5_Dest and WSF5_Origin denoted the maximum sustained wind speed at the destination and origin airports, respectively. High wind speeds could impact flight operations, potentially leading to delays.

The historical performance of the previous airport (Previous_Airport) from which the aircraft arrived before the current flight could also have had an impact on the subsequent flight's arrival time. This highlighted the

interconnectedness of flight operations and the importance of considering the overall performance of the aviation system.

Other features, such as Plane_Age, which indicated the age of the aircraft operating the flight, could have contributed to delays. Older aircraft might have been more prone to mechanical issues, which could have caused delays. The number of seats (Number_Of_Seat) on the aircraft could also have impacted arrival delays, as larger aircraft might have required more time for boarding and deboarding.

Additionally, features related to airport services and resources, such as Avg_Monthly_Pass_Airport,Avg_Monthly_Pass_Airline, Ground_Serv_Per_pass, and Flt_Attendants_Per_Pass, provided insights into passenger volumes, ground handling efficiency, and service quality. These factors could indirectly affect arrival delays.

Finally, weather-related features played a significant role, other factors also notably impacted arrival delays. The 51 features identified through the Random Forest feature importance analysis on aircraft arrival delays provided a comprehensive understanding of the numerous influencers affecting the timeliness of aircraft arrivals. These findings highlight the essential additional data that should be integrated into the model for a more thorough analysis. Considering these factors, airlines, airports, and aviation stakeholders can identify areas for improvement and implement strategies to minimize arrival delays, ultimately enhancing the efficiency and reliability of air travel.

SHAP values

The SHAP values derived from the Random Forest Method underscore the varying contributions of distinct features in forecasting flight delays across Class 0 (not delayed) and Class 1 (delayed) flights, as depicted in Figure 25

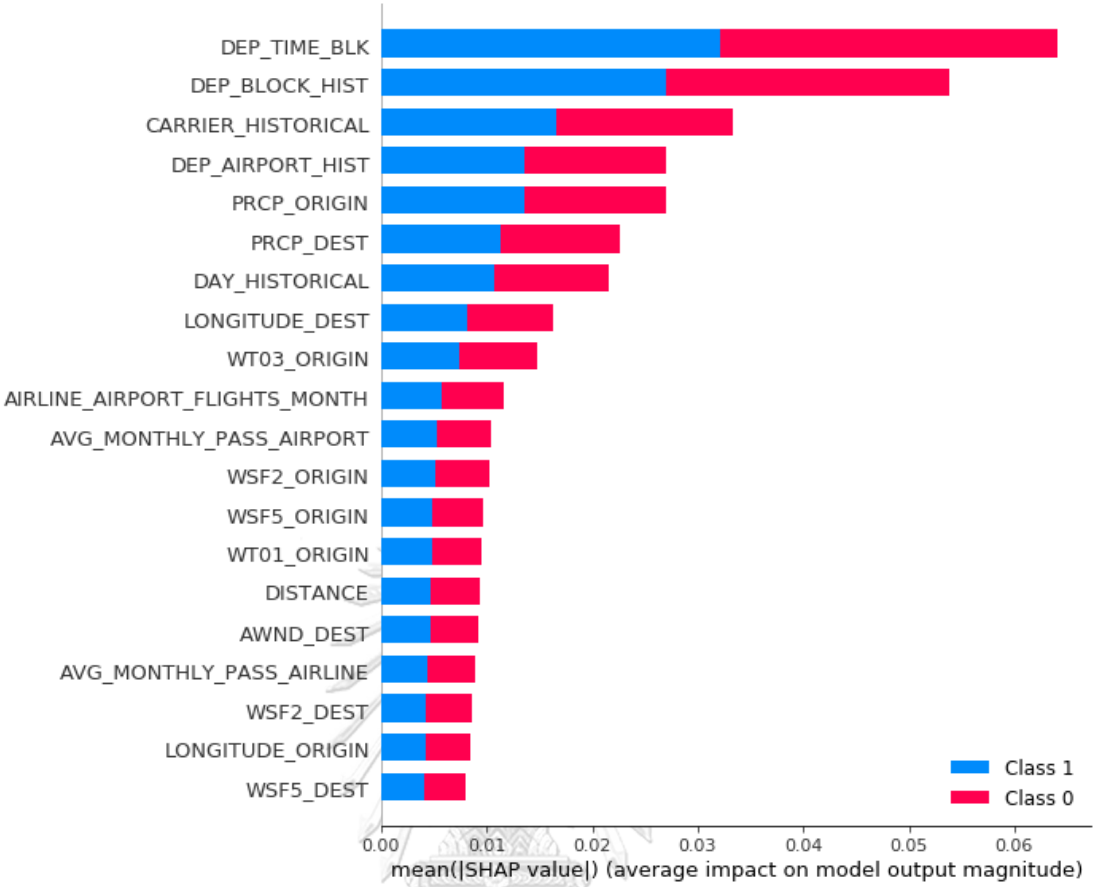


Figure 25: SHAP values Random Forest Method

Dep_time_blk: This feature holds a SHAP values of 0.03 for Class 1 (delayed flight) and 0.06 for Class 0 (non-delayed flights), indicating a stronger influence on identifying non-delayed flights compared to delayed ones.

Dep_block_hist: With Class 1 at 0.027 and Class 0 at 0.056, this feature exhibits a more substantial impact on predicting non-delayed flights than delayed ones, reflecting historical departure block patterns' significance.

Carrier_historical: Class 1 at 0.015 and Class 0 at 0.037 reveal that carrier historical data plays a more significant role in predicting non-delayed flights than delayed ones.

Dep_airport_hist: Displaying values of 0.012 for Class 1 and 0.026 for Class 0, this feature's higher influence on Class 0 suggests a stronger role in predicting non-delayed flights based on historical departure airport patterns.

Overall, these SHAP values depict distinct impacts of features on flight delay predictions. Historical departure block patterns, carrier history, and departure airport history significantly influence predicting non-delayed flights (Class 0)

over delayed flights (Class 1), offering valuable insights into feature importance for the Random Forest's predictions in both classes.

Aviation enthusiasts or interested individuals seeking aviation-related data like METAR (Meteorological Aerodrome Report), TAF (Terminal Aerodrome Forecast), and surface observations can access these details from various user-friendly online platforms. Websites like Aviation Weather Center (aviationweather.gov) offer comprehensive resources, providing METAR and TAF reports for specific airports globally, along with graphical weather analyses and forecasts. The National Weather Service's website (weather.gov) also offers easy access to METAR and TAF reports, allowing ordinary individuals to retrieve current weather conditions and forecasts for airports worldwide. For a more user-friendly interface, websites such as FlightAware and SkyVector provide access to METAR and TAF data, along with flight tracking services, making it convenient for aviation enthusiasts or travelers to access real-time weather information and surface observations for flight planning and general weather awareness. These platforms offer user-friendly interfaces and easy navigation, enabling ordinary people to access valuable aviation-related data efficiently.

Another perspective to consider is how the airline could utilize its routes to make up for lost time. Figure 26 displays the wind aloft speed and direction at different altitudes, requiring the pilot to make decisions about the most advantageous altitude to cruise at. For example, headwinds would slow the aircraft down, while tailwinds would speed up the aircraft's ground speed.

Wind/Temp Text (FB Winds)

The Winds/Temps text data are the official FB Winds product from NCEP. This provides winds from 3000 to 53,000 feet out to 6, 12 and 24 hours. The data is extracted from model output for individual airports represented by a 3 letter IATA identifier. The data are grouped by region. Here is a sample:

```
DATA BASED ON 061800Z
VALID 070000Z   FOR USE 2000-0300Z. TEMPS NEG ABV 24000

FT  3000    6000    9000   12000   18000   24000  30000  34000  39000
BRL 2313 2512+02 2811-02 3116-09 3336-22 3546-33 355949 355659 312855
DBQ 2122 2413+02 2715-02 3021-09 3334-21 3646-33 345849 345459 323256
DSM 2128 2319+03 2817-01 3020-06 3230-18 3333-32 334649 346958 333660
MCW 2136 2427+04 2627-01 2821-07 3131-19 3237-32 325050 337359 324359
```

The data block shows ddffttt where dd is the wind direction in 10s of degrees, ff is the win speed in knots and ttt is the temperature in Celsius. At higher levels, the sign for the temperature is not needed since it's assumed all values are negative. If the first digit of the wind direction is greater than 4, then the wind speed is greater than 100. For example, 810550 would be 310 degrees (80-50) at 105 knots and the temperature is -50C.

Figure 26: Winds aloft chart (Aviation Weather Center, 2022)

The real-time weather data would greatly help the pilot determine the contingency plans for both the air crew and the ground crew to reduce the turnaround time. If possible, more data for aviation weather are needed to be

incorporated into the flight data. Specific information on aviation weather, especially surface weather, is important for aircraft delays.

The surface weather included wind speed, temperature, and visibility. There were terminal area forecasts as well as other weather information that was local near the airport. The current set of data was collected throughout the year. Future studies needed to collect specific data for specific weather seasons. The distribution of weather data and their types was very important. Most of the time, the aircraft was on time. If the weather was not suitable to fly, then the flight mostly did not happen.

The weather data is necessary to accurately plan for preventive maintenance. This could decrease downtime, increase aircraft availability, and maintain optimal spare parts storage. The longer the aircraft stayed in service, the more opportunities for the airline to make profits.

Chapter 5: Conclusion

In the pursuit of uncovering efficient solutions to predict and mitigate aircraft delays, this study extensively relied on datasets sourced from Kaggle. Employing a comprehensive approach, four prominent machine learning methodologies—Random Forest, CatBoost, Gradient Boosting, and AdaBoost—were rigorously evaluated. Leveraging the powerful GridSearchCV, a systematic exploration of hyperparameters was conducted to identify the most optimal combinations. Throughout this investigation, accuracy metrics and F1-scores were employed as pivotal benchmarks, serving as robust indicators to determine the most effective machine learning method. These metrics not only gauged the predictive capabilities but also facilitated the selection of the most adept model for addressing the complexities associated with aircraft delays in the aviation industry.

Considering the results and factors for future model improvement, several recommendations can be made:

Random Forest: with an accuracy rate of 83% and an F1-score of 0.56, Random Forest stands out as a strong candidate for prediction models. To further enhance its performance, hyperparameter tuning is crucial. Exploring different combinations of estimators, max depth, and max features can optimize its predictive capabilities. Additionally, employing feature engineering techniques can enhance the relevance and quality of input features.

Gradient Boosting: with an accuracy of 81.1% and an F1-score of 0.47, and a higher number of estimators than other algorithms, Gradient Boosting holds its own. Fine-tuning the learning rate, exploring various max feature settings, and optimizing the max depth can potentially elevate its performance. Furthermore, feature engineering and interpreting the model's decisions can provide further insights and improvements.

CatBoost: with an accuracy of 81% and an F1-score of 0.46, CatBoost exhibits promise in handling categorical features effectively. To improve its performance, adjustments to the learning rate, exploration of different boosting algorithms, hyperparameter tuning, and feature engineering can be advantageous.

AdaBoost: Despite its lower accuracy of 72.6% and an F1-score of 0.45, AdaBoost shows potential for improvement in prediction models. Fine-tuning the learning rate, exploring various ensemble configurations, and addressing

class imbalances can elevate its performance. Feature engineering and addressing class imbalances, if applicable, can contribute to better results.

To enhance future predictions, the recommendation is to collect more available data and utilize real-time weather sensors for higher accuracy. Gathering additional information from winds aloft and local airport winds, updated hourly or as necessary, plays a pivotal role in determining whether a pilot will proceed with the flight or make the crucial decision to delay the aircraft due to adverse weather conditions.

For an advanced version of the model, exploring a combination of methods, such as merging the Random Forest method with AdaBoost or decision trees, is a viable approach. Comparing the outcomes of these combined classification models will help determine the effectiveness of this new information or combination.

By incorporating these recommendations and further refining the model, the aim is to achieve more accurate predictions and enhance decision-making processes.

The aviation industry contends with substantial delays, stemming from a myriad of factors spanning adverse weather conditions like thunderstorms, snow, fog, and strong winds. These conditions not only pose safety risks but disrupt meticulously planned flight schedules, creating widespread operational challenges. Congestion on runways and limitations in airport infrastructure exacerbate delays, particularly during peak periods, intensifying operational complexities. Aircraft maintenance issues, technical glitches, and unforeseen incidents further compound airlines' challenges. Weather stands out prominently among these factors, emphasizing the urgent need for specialized predictive models to forecast and manage weather-related flight delays, crucial for navigating operational hurdles in the aviation sector. The comprehensive state of the aviation system, encompassing air traffic management, weather forecasting, and maintenance protocols, significantly influences addressing and mitigating delays. The integration of advanced technologies and predictive analytics is pivotal in optimizing air traffic routes, furnishing precise weather data, and refining aircraft maintenance processes. Implementation of these solutions is imperative for curbing aircraft delays and augmenting the overall efficiency and dependability of air travel.

REFERENCES

America, A. F. (2022). U.S. Passenger Carrier Delay Costs. Retrieved from https://www.airlines.org/dataset/u-s-passenger-carrier-delay-costs/

Bouwer, J., Vik Krishnan, Steve Saxon, & Tufft, C. (2022). Taking stock of the pandemic's impact on global aviation. Retrieved from https://www.mckinsey.com/industries/travel-logistics-and-infrastructure/our-insights/taking-stock-of-the-pandemics-impact-on-global-aviation.#/

Brueckner, J. K., Czerny, A. I., & Gaggero, A. A. (2021a). Airline mitigation of propagated delays via schedule buffers: Theory and empirics. *Transportation Research Part E: Logistics and Transportation Review*, *150*, 102333. doi:https://doi.org/10.1016/j.tre.2021.102333

Brueckner, J. K., Czerny, A. I., & Gaggero, A. A. (2021b). Airline schedule buffers and flight delays: A discrete model. *Economics of Transportation, 26-27*, 100218. doi:https://doi.org/10.1016/j.ecotra.2021.100218

BTS. (2023). Flight Delays by Cause. Retrieved from https://www.transtats.bts.gov/OT_Delay/ot_delaycause1.asp?6B2r=FE&20=E.

Dissanayaka, D. M. M. S., Adikariwattage, V., & Pasindu, H. R. (2019, 2019/10). *Evaluation of Emissions from Delayed Departure Flights at Bandaranaike International Airport (BIA)*. Paper presented at the Proceedings of the 11th Asia Pacific Transportation and the Environment Conference (APTE 2018).

FAA. Delay Propagation. Retrieved from https://aspm.faa.gov/aspmhelp/index/Delay_Propagation.html

FAA. Handbooks & Manuals. Retrieved from https://www.faa.gov/regulations_policies/handbooks_manuals., from Federal Aviation Administration https://www.faa.gov/regulations_policies/handbooks_manuals.

FAA. Types of Delay. Retrieved from https://aspm.faa.gov/aspmhelp/index/Types_of_Delay.html

FAA. (2020). *Air Traffic by the Numbers*. Retrieved from https://www.faa.gov/air_traffic/by_the_numbers/media/Air_Traffic_by_the_Numbers_2020.pdf

FAA. (2022a). Air Traffic by the Numbers.

FAA. (2022b). *FAA Aerospace Forecast FY 2022-2042*(pp. 144). Retrieved from https://www.faa.gov/sites/faa.gov/files/2022-06/FY2022_42_FAA_Aerospace_Forecast.pdf

FAA. (2022c). FAQ: Weather Delay. Retrieved from https://www.faa.gov/nextgen/programs/weather/faq

FAA. (2023). *Aeronautical Information Services Aeronautical Chart Users' Guide*(pp. 137). Retrieved from https://aeronav.faa.gov/user_guide/20231005/cug-complete.pdf

GAO. (2021). Observations on the Ongoing Recovery of the Aviation Industry. *United States Government Accountability Office*(GAO-22-104429). Retrieved from https://www.gao.gov/assets/gao-22-104429.pdf

Gui, G., Liu, F., Sun, J., Yang, J., Zhou, Z., & Zhao, D. (2020). Flight Delay Prediction Based on Aviation Big Data and Machine Learning. *IEEE Transactions on Vehicular Technology*, *69*, 140-150.

Hajko, J., & Badánik, B. (2020). Airline on-time performance management. *Transportation Research Procedia, 51*, 82-97. doi:https://doi.org/10.1016/j.trpro.2020.11.011

Hu, P., Zhang, J., & Li, N. (2021, 20-22 Oct. 2021). *Research on Flight Delay Prediction Based on Random Forest.* Paper presented at the 2021 IEEE 3rd International Conference on Civil Aviation Safety and Information Technology (ICCASIT).

ICAO. (2021). 2021 global air passenger totals show improvement from 2020, but still only half pre-pandemic levels Retrieved from https://www.icao.int/Newsroom/Pages/2021-global-air-passenger-totals-show-improvement.aspx

Kafle, N., & Zou, B. (2016). Modeling flight delay propagation: A new analytical-econometric approach. *Transportation Research Part B: Methodological, 93*, 520-542. doi:https://doi.org/10.1016/j.trb.2016.08.012

Kasirzadeh, A., Saddoune, M., & Soumis, F. (2017). Airline crew scheduling: models, algorithms, and data sets. *EURO Journal on Transportation and Logistics, 6*(2), 111-137. doi:https://doi.org/10.1007/s13676-015-0080-x

Khaksar, H., & Sheikholeslami, A. (2019). Airline delay prediction by machine learning algorithms. *Scientia Iranica, 26*(5), 2689-2702. doi:10.24200/sci.2017.20020

learn, s. Sklearn preprocessing OrdinalEncoder. Retrieved from https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.OrdinalEncoder.html

learn, s. Sklearn preprocessing StandardScaler. Retrieved from https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html

Liu, F., Sun, J., Liu, M., Yang, J., & Gui, G. (2020). *Generalized Flight Delay Prediction Method Using Gradient Boosting Decision Tree*.

Mueller, E., & Chatterji, G. (2002). Analysis of Aircraft Arrival and Departure Delay Characteristics. In *AIAA's Aircraft Technology, Integration, and Operations (ATIO) 2002 Technical Forum*.

Wu, C.-L. (2005). Inherent delays and operational reliability of airline schedules. *Journal of Air Transport Management, 11*(4), 273-282. doi:https://doi.org/10.1016/j.jairtraman.2005.01.005

Zou, B., & Hansen, M. (2012). Flight delays, capacity investment and social welfare under air transport supply-demand equilibrium. *Transportation Research Part A: Policy and Practice, 46*(6), 965-980. doi:https://doi.org/10.1016/j.tra.2012.02.015

VITA

NAME	Patara Charnvanichborikarn
DATE OF BIRTH	16 March 1992
PLACE OF BIRTH	Bangkok
INSTITUTIONS ATTENDED	Sasin School of Management
HOME ADDRESS	336 Moo4 Watcharapol Rd. Tharaeng Bangkhen Bangkok 10220 Thailand