Aalto University
School of Science
Master’s Programme in Information Networks

Eppu Halmesaari

Interpretable machine learning for prediction of aircraft turnaround times

An explainable aggregated approach

Master’s Thesis

Helsinki, October 30, 2020

Supervisor: Jukka Luoma, Assistant professor

Thesis advisor(s): Mika Tiilikainen, Lic.Tech, E.E.

Author Eppu Halmesaari

Title of thesis Interpretable machine learning for prediction of aircraft turnaround times

Programme Information Networks

Major ICT In Business

Thesis supervisor Assistant professor Jukka Luoma

Thesis advisor(s) Mika Tiilikainen, Lic.Tech, E.E.

Collaborative partner Finnair

Date 30.10.2020 **Number of pages** 100 + 6 **Language** English

Abstract

Optimized aircraft turnaround processes are crucial for airlines to maximize the utilization of single aircrafts, but overly optimized scheduled turnaround durations may compromise schedule adherence. Predictions about the duration and information related to the drivers of the turnaround duration provide operatively valuable information as they enable reactive measures to be taken to mitigate possible delays. In addition, by revealing general patterns and potential bottlenecks in a turnaround process, this information can be used to improve individual subprocesses in the longer term.

This thesis presents a proof-of-concept of an operative model which creates interpretable predictions of single turnarounds by showing the contributions of different factors on the predicted turnaround duration. Main elements of the model are the tree-based machine learning algorithm XGBoost and interpretability framework SHAP. Explanations are provided both on the local and global level which means that they can be used to interpret individual turnarounds and to deduce general patterns between variables in the dataset.

Dataset of this study consists of variables such as passenger count and the amount of cargo which are used to describe different subprocesses. However, explicit data about the durations of subprocesses is not included in the dataset. From a scientific perspective the combination of such dataset and the methods mentioned, this thesis provides a novel explainable aggregated approach on the turnaround duration prediction by viewing the turnaround as a black box and by inferring the effects of different variables on the total duration.

Results of this study imply that an explainable aggregated approach is capable of explaining the turnaround duration to a certain extent. The proof-of-concept model enables the deduction of the main patterns in the data and also provides useful information related to individual turnarounds. However, additional features are needed to be able to provide a more thorough understanding of the turnaround process in this kind of approach.

Keywords turnaround, aggregated approach, predictive modeling, interpretability

Tekijä Eppu Halmesaari

Työn nimi Selittävä koneoppiminen lentokoneen kääntöaikojen ennustamiseen

Koulutusohjelma Information Networks

Pääaine ICT In Business

Vastuuopettaja/valvoja Apulaisprofessori Jukka Luoma

Työn ohjaaja(t) TkL Mika Tiilikainen

Yhteistyötaho Finnair Oyj

Päivämäärä 30.10.2020 **Sivumäärä** 100 + 6 **Kieli** Englanti

Tiivistelmä

Optimoitu lentokoneiden kääntöaika mahdollistaa yksittäisten lentokoneiden suuremman käyttöasteen, mutta liialti optimoitu kääntöaika voi vaarantaa lentokoneiden aikataulussa pysymisen. Näin ollen ennusteet yksittäisten kääntöjen kestoista sekä tieto kääntöprosessin merkittävimmistä ajureista mahdollistavat myöhästymistä ehkäisevät toimenpiteet. Lisäksi ymmärrys yleisistä riippuvuussuhteista ja pullonkauloista käännön alaprosessien välillä mahdollistaa prosessien kehittämisen pidemmällä aikavälillä.

Tämä työ esittelee operatiivisen työkalun mallin, joka luo selittäviä ennusteita yksittäisistä lentokoneen kääntöajoista tuomalla esiin yksittäisten selittävien tekijöiden vaikutuksen lopullisessa ennusteessa. Mallin pääosat ovat koneoppimisen algoritmi XGBoost sekä selittävyyttä tuova menetelmä SHAP. Selittävyyttä kääntöprosessin kestoon tuodaan sekä yksittäisen käännön tasolla että koko käytössä olevan datan tasolla. Näin ollen on mahdollista tutkia myös yleisiä riippuvuussuhteita muuttujien välillä.

Käytettävissä oleva data sisältää muuttujia, joiden välityksellä pyritään kuvailemaan yksittäisiä käännön alaprosesseja, mutta tarkkaa tietoa yksittäisten alaprosessien kestosta ei ole saatavilla. Tieteellisestä näkökulmasta tämän tyyppisten menetelmien ja datan yhdistäminen luo uniikin lähestymistavan kyseiseen ongelmaan. Lähestymistapaa voidaan kutsua selittäväksi aggregaattimalliksi, jossa itse kääntöprosessin keston määräytymistä kohdellaan mustana laatikkona ja jonka käyttäytymistä pyritään selittämään erilaisilla muuttujilla.

Tulosten perusteella selittävä aggregaattimalli kykenee antamaan selittäviä ennusteita käännön kestosta. Malli tuo esille hyödyllistä tietoa kuvailemalla yksittäisen käännön merkittävimpiä tekijöitä ja lisäksi se tuo esille yhteyksiä muuttujien välillä. Tulosten perusteella voidaan myös todeta, että tässä työssä käytettävissä olevat muuttujat kykenevät selittämään ilmiötä vain rajallisesti ja esimerkiksi paremman ennustustarkkuuden saavuttaminen vaatii uusia muuttujia ja dataa.

Avainsanat aggregaattimalli, ennustava mallintaminen, selittävyys, koneoppiminen, lentokoneen kääntö

Esipuhe

Vuosi 2020 jää muistoihin erikoisena vuotena monista syistä, mutta ennen kaikkea siitä, että saan päätökseen jo vuodesta 2009 jatkuneen yliopistotaipaleeni! Suuret kiitokset kuuluvat valvojalleni Jukka Luomalle, joka auttoi tutkimuksen tieteellisen näkökulman teroittamisessa sekä ohjaajalleni Mika Tiilikaiselle, joka antoi arvokkaita ohjeita analyysin tekemiseen. Molempien opeista on taatusti hyötyä myös jatkossa. Kiitokset myös Henri Kalsille, joka auttoi datan haalimisessa ja osallistui työn eri vaiheiden läpikäymiseen. Lopuksi kiitokset vielä vanhemmille tuesta pitkällä koulutiellä sekä Magdalle arvokkaista kommenteista ja tuesta.

Helsinki, 30.10.2020

Eppu Halmesaari

Table of Contents

1 Introduction

In aviation industry there exists a common impression that airplanes earn revenue only when they are airborne (Wang, Wang & Wu, 2017). From an airline perspective this means that the time spent for flight operations transporting passengers and cargo should be maximized and the time spent for ground operations minimized. Turnaround is a part of ground operations of an aircraft and it encompasses all the activities made to prepare an arriving airplane for a new outbound flight leg (Wu, 2008). Optimized turnaround process with a short duration is therefore important for an airline to be able to maximize the utilization of a single aircraft.

According to Oreschko et al. (2012), the subprocesses of a turnaround such as cargo loading, passenger boarding, catering and refuelling have a stochastic nature leading to variations in their duration and variation in the total duration of a turnaround. The time that is reserved for a single turnaround affects the schedule of an airline as airplanes need to be able to depart according to their predetermined schedules (Oreschko, Schultz, Elflein & Fricke, 2010). Due to the stochastic nature of a turnaround and the need for schedule adherence, there exists a trade-off where short turnarounds are required for maximal utilization of an airplane, but at the same time there is danger that the departure of the next flight leg is delayed due to varying turnaround lengths.

For airlines, the trade-off means there is a need for realistic estimation of the length of a turnaround to be able to set a schedule that is short but contains enough buffer time to absorb possible fluctuations in subprocesses of a turnaround without compromising the departure schedule. The problem is that usually airline schedule planning takes place months before the flight is actually performed (Lohatepanont & Barnhart, 2004) but the factors that affect the turnaround time, such as weather or passenger number, are determined latest at the same day as the turnaround is supposed to take place. In addition to lengths of subprocesses such as boarding or fueling, actual available turnaround time may be shorter than the scheduled turnaround time if aircraft arrives late from its previous flight (Oreschko et al., 2012). This results in a situation where accurate estimate of the actual turnaround duration is valuable even until the end of the turnaround process as airlines may adjust their operations such as decide for which subsequent operations the aircraft can be used or how much time is available for passengers who have connecting flights (Clarke, 1998).

In this thesis, a quantitative analysis based on a dataset collected from aircraft turnarounds in a single airport is performed. Airline operator providing the information is Finnair and the dataset spans flights from a period of several years.

After the exploratory data analysis and preprocessing of data, a tree boosting machine learning algorithm called XGBoost is applied on the data to produce a tool to predict future turnaround durations. In addition to numerical predictions, statistical methods together with a framework based on the calculation of Shapley values (SHAP) are used to provide interpretability both in global and local level of the dataset.

By breaking individual predictions into factors where the contribution of each independent variable is shown, it is possible to evaluate reliability of those predictions. In addition, global trends of independent variables in the dataset may reveal operationally beneficial information which can be utilized in a longer time span. Thus, this research attempts to provide a proof-of-concept of an operative tool which enables stakeholders to gain information about the performance of a coming turnaround and enable them to react based on the information received both in short and longer term. Compared to previous scientific research, this study provides a novel explainable aggregated approach by inspecting the turnaround as a black box model and demonstrating explainable predictions without explicitly modeling the subprocesses. Instead, information regarding the subprocesses is provided through explicit input parameters such as passenger count.

The section 2 of this thesis will focus on the background related to the turnaround process as well as discuss the on-time performance and value of predictions related to a turnaround process. In addition, section 2 will provide overview of previous research related to aircraft turnaround predictions. In section 3, the research objective and research questions will be formulated based on the information provided by section 2. Section 4 focuses on the methodology part which includes further analysis of machine learning methods and interpretability models, introduction of data and data analysis as well as the detailed introduction to selected models for the analysis.

2 Background & previous research

In this section the background and dynamics related to an aircraft turnaround and its subprocesses and possible variations are reviewed. The description of a turnaround process is followed by a literature review based on current research about modelling and prediction of turnaround processes as well as prediction of aircraft delays in general.

2.1 Aircraft turnaround process

According to International Air Transport Association (IATA), turnaround of an aircraft comprehends the time period when an aircraft occupies an apron stand or a gate in an airport, and in practice turnaround time is the time between on block and off block times (arrival at the gate and start of pushback/taxi) (Oreschko et al., 2010). Turnaround is finished when all doors of the aircraft are closed, boarding bridge or stairs are removed, pushback vehicle is available and aircraft is in general ready for pushback (Nosedal Sanchez & Piera Eroles, 2018). Schmidt (2017) lists that turnaround process consists of different subprocesses where the most typical ones are passenger deboarding and boarding, cargo unloading and loading, catering, waste tank and potable water service, cleaning, and refuelling. Procedures such as a walk around, which is performed by the flight crew to check airworthiness of the aircraft or the emergency equipment checks performed by the cabin crew, are also part of the turnaround process.

De-icing is an example of a turnaround subprocess that is not performed by default but only if environmental factors lead to a situation where de-icing is necessary and there exists a procedure to perform it on-stand (Norin, Granberg, Yuan & Värbrand, 2012). In this thesis, de-icing performed on a dedicated de-icing area where aircraft needs to move by using its own power source is not considered as part of the turnaround. Due to the multitude of different subprocesses belonging to a turnaround, they may be provided by more than one service provider (Wu & Caves, 2004b).

Some of the subprocesses in a turnaround can be performed in parallel, such as cargo loading and cleaning, whereas procedures, such as cleaning and boarding, need to be performed in sequence (Schmidt, 2017). Requirements for the turnaround process flow may be due to legal or logistical aspects (Oreschko et al., 2010). It is evident that cleaning of the aircraft cabin is not possible when passengers are sitting on their designated seats, but on the other hand, the sequence of performing boarding and refuelling is not always the same. In some cases, such as refuelling, precautionary procedures may allow parallel operation of certain turnaround processes (Regulation 859/2008, 2008). Refuelling is normally performed when passengers are not onboard the aircraft. When precautionary procedures are carried out, refuelling is allowed to be performed simultaneously with deboarding and boarding.

Critical path in aircraft turnarounds is the longest chain of subprocesses which need to be performed in sequence and therefore limit the minimum turnaround time (Oreschko et al., 2010). According to Fricke and Schultz (2009), deboarding, boarding, cabin preparation and refuelling usually constitute the critical path whereas unloading, loading and aircraft servicing are not on the critical path.

As mentioned in the introduction, turnarounds and their subprocesses have a stochastic nature due to variations in the magnitude of flight related variables. The number of passengers travelling inbound or outbound, as well as the amount of cargo and baggage onboard, change between flights (Wu, 2008). More passengers or cargo means longer boarding and loading times. Same applies to refuelling need, which depends on the length of the flight, weight of the aircraft, and whether the amount of fuel includes the fuel to be burned by the return flight (tankering) (Guerreiro Fregnani, Müller & Correia, 2013). Stochasticity of turnaround processes means that the criticality of different subprocesses between different turnarounds may vary.

According to Schmidt (2017), turnaround time is also dependent on the aircraft type and the business model of the airline. Each aircraft type has a certain capability in terms of weight and seat capacity. Also, the cabin layout or dimensions such as aisle width that furthermore affect the time needed for processes such as deboarding, boarding or cabin cleaning, vary between aircraft types (Wang et al., 2017). Business model, on the other hand, determines what kind of service level is provided in terms of catering and cleanliness of the aircraft and therefore has an effect on the corresponding subprocess duration (Lawton, 2003).

Turnarounds can take place either at the gates or at apron stands (Schmidt, 2017). In apron stands passengers need to deboard and board the aircraft by using integrated or separate stairs. From apron stands passengers are transported onwards to a terminal. This difference between a gate position or an apron stand might affect the duration of the deboarding and boarding process (Diepen, Pieters, Van Den Akker & Hoogeveen, 2013). Other airport dependent factors that might affect the length of a turnaround is the skill level of the ground staff that may vary between airports (Oreschko et al., 2012) and manpower or equipment availability (Fricke & Schultz, 2009).

Schultz and Reitmann (2019) note that ground handling processes are mostly executed by professionals whereas boarding is also affected by the willingness and ability of the passengers. They also mention that in addition to number of passengers and passenger behavior, the high number of hand luggage or amount of priority passengers may affect the boarding process. Schmidt (2017) points out that the boarding strategy and number of doors used for boarding affect the boarding duration. In this study however, the boarding strategy and number of doors used for boarding can be considered constant between turnarounds.

From a weather perspective especially environmental conditions that would result in a need for de-icing may have an effect on the total turnaround length (Norin et al., 2012). Wang et al. (2017) point out that in the determination of turnaround time low visibility and bad weather in general should be taken into account. This might be due to the fact that extremely low visibility or bad weather have an overall slowing effect on airport operations.

The factors which have an effect on different turnaround subprocesses are presented above. These may have an effect on the total length of a single turnaround. The factors can be divided into four groups which are airport specific factors, weather related factors, aircraft specific factors and factors related to the general flow of people and material. It is worth to note that most of the information related to these factors is available before a flight is conducted and therefore can be anticipated in a prediction algorithm with correct parameters. In addition to factors related to normal operation of an aircraft during a turnaround, different kinds of disruptions may affect a turnaround process. Wu (2008) discusses that disruptions may happen due to connecting passengers, connecting crew, missing check-in passengers, late inbound cargo or baggage and due to equipment breakdown. Especially technical issues that belong to the Minimum Equipment List, meaning that flight dispatch is not possible until such problems are resolved, may cause significant delay (Fricke & Schultz, 2009). Analysis and anticipation of disruptions mentioned is out of the scope of this study, even though they could possibly be modelled in some way.

2.2 Scheduled and available turnaround time

In airline schedule planning a tradeoff exists. According to Wu (2010), a long turnaround time reduces the utilization of aircraft as the time spent on ground could be spent on revenue making in the air. At the same time, long turnaround time stabilizes airline operations as it helps in managing complex turnaround processes and reduces delays which in turn can cause costs. Therefore, a scheduled turnaround time is a compromise between these two factors and there exists a motivation for an airline to keep the scheduled turnaround as short as possible to enable efficient aircraft use.

Airline schedules are usually determined many months before the actual flight takes place and the schedules are partly based on history data (Lohatepanont & Barnhart, 2004). Aircraft manufacturers provide baseline values for the servicing of the aircraft such as boarding rate for passengers or loading rate for cargo (Airbus, 2020; Boeing, 2005). However, as discussed, actual turnaround duration is determined by the length of its subprocesses, which contain variations such as changing passenger number or aircraft type and such information is only available closer to the actual flight (Oreschko et al., 2012). Therefore, scheduled turnaround time is, at its best, an estimate of the actual turnaround time.

Due to the stochasticity of turnaround durations and varying daily turnaround related parameters, the realized turnaround duration can be shorter or longer than the scheduled turnaround time. In addition, as Oreschko et al. (2012) point out, scheduled turnaround time can also be affected by late arrival of the aircraft from a previous flight leg. In such a case, scheduled turnaround time is already shortened by the amount of arrival delay and from operative perspective it is more suitable to compare the actual turnaround to the available turnaround time where available turnaround time starts from the time of the actual arrival and is constrained by the scheduled departure time. If the actual turnaround time is greater than the available turnaround time, the flight will have a late departure.

2.3 Aircraft delays and on-time performance

Aircraft turnaround affects many stakeholders as it has a direct impact to airport, ground handling, air traffic management and air traffic controller resources (Nosedal Sanchez & Piera Eroles, 2018). In addition, a commercial flight requires a crew, an aircraft and passengers or cargo for the flight to be performed. According to Wu (2010), airline operations can be considered as a complex network which includes both a spatial and a temporal aspect where schedule adherence is important. In practice, in transportation of passengers and cargo, a certain schedule is expected to be followed and the contribution of all stakeholders is needed to achieve on-time performance which is crucial for airline profitability.

Delays cost money for airlines and passengers (Wu, 2008) and may also disrupt the economic utilization of airport capacity (Oreschko et al., 2010). Wu and Caves (2004a) note that delays cause lost time for passengers and may lead to customer disloyalty and loss of goodwill as well as result in compensation requests. They (Wu & Caves, 2004a) also point out that schedule disruptions require extra resources from airline operations as they need to coordinate possible changes in aircraft, crew, cargo or passenger connections. Turnaround times that lead to aircraft departure delay may lead to missed time slots at the airport which consequently may lead to even longer departure delay (Jaehn & Neumann, 2015). It is important to note that due to the nature of resource exchange in flight operations, delays occurred somewhere in the chain of events may propagate further through the network of operations (Wu, 2010).

In contrast to delay costs, Wu and Caves (2004a) list advantages from high turnaround punctuality which are maximal fleet utilization, minimized operational disturbance in terminals, maximized utilization of airline resources and more efficient airport utilization of airport facilities. As a summary we can conclude that benefits from schedule punctuality in turnarounds for different stakeholders are undeniable. Even small delays that are indifferent for a single flight might propagate further down in the chain of operations and become a critical factor for another individual operation. Therefore, the mitigation of such delays in flight operations is important in daily airline operations.

As mentioned in the description of the turnaround process, there can be many reasons for a long duration of a turnaround belonging to normal variations in the subprocesses such as large number of passengers or a large amount of fuelling need. In addition, duration of a turnaround can also be long due to disruptions such as late connecting passengers or technical problems. Whatever the cause for a lengthened turnaround process is, in the event of a departure delay the costs of not adhering to a schedule might be realized. In this thesis, the variations of normal turnaround related parameters such as passenger number and amount of cargo or de-icing need are used to determine the turnaround time.

Next, the value of accurate predictions related to turnaround durations and the tactical methods to reduce possible delays based on those predictions will be discussed. The industry standard for definition of late arrival or departure is based on the 15 minutes difference to original schedule (Wu, 2010). However, in this thesis an arrival or departure is considered to be delayed whenever the actual arrival or departure happens after the corresponding scheduled arrival or departure time.

2.4 Reactive measures against delays and the value of predictions

Compared to flight phases or other ground processes, airlines have a better control and flexibility over the turnaround process, in terms of tactical measures to mitigate delays and delay related costs (Wu, 2010). In general, when comparing ground phases to flight phases, uncertainties in different flight phases have less impact on the schedule punctuality than in ground phases (Schultz & Reitmann, 2019).

The set of possible preventive actions to avoid flight delay depends on how long in advance the information on possible delays is available. Longest time span is multiple months when airlines perform schedule planning and have the chance of providing extra buffer to turnaround times for frequently delayed flights (Kohl, Larsen, Larsen, Ross & Tiourine, 2007). Closer to the actual flight airlines have released their schedules but they still have the chance to inform passengers about possible delays to minimize monetary loss and also the inconvenience that is caused for the passengers (Choi, Kim, Briceno & Mavris, 2017). According to Horiguchi et al. (2017), until several days before the flight, a suitable gate can be assigned or more resources for the ground handling team can be deployed. More resources can shorten the turnaround time by speeding up subprocesses such as loading and unloading of bags, cargo or catering, required that they are on the critical path of the turnaround.

Until the flight actually occurs, flight crew can be instructed to shorten the time spent for fuelling which may be possible through precautionary procedures (Regulation 859/2008, 2008; Horiguchi et al., 2017). Turnaround operations can also be accelerated by omitting cargo loads or delaying them to later flights which applies also for baggage. However, in the case of moving baggage transportation to a new flight, the cost of baggage delivery to the final destination needs to be less than the cost of the flight delay (Wu, 2010).

During enroute phase of the aircraft operations there is a possibility to reduce accumulated or suspected delay by flying with higher cruise speeds according to Wu (2010). This causes extra fuel burn and such cost needs to be outweighed by the cost of the delay, he notes. Compared to short-haul operations, during long-hauls operation there is a better chance to absorb possible delay due to longer flight times (Schmidt, 2017).

As a summary, it can be said that multiple ways exist to mitigate accumulated delay or prevent upcoming delay from taking place. Different means to cope with delay have different time spans when they can be applied. What is common for all the different means to prevent upcoming delay from happening is that they rely on information and predictions about future events. This means that accurate predictions are valuable for operative decision making as delay related costs can be avoided with correct preventive actions. Accurate predictions on the other hand are dependent on accurate information which is usually available closer to the flight event as more information is available (Oreschko et al., 2012).

In this thesis, the prediction algorithm requires input parameters such as number of passengers, amount of cargo and aircraft type. This kind of information can be considered to be available few days before the scheduled departure for such extent that possible predictions can be given. However, information such as late arrival from previous leg, which is related to the information regarding available turnaround time, is available latest until the beginning of the flight. Therefore, the prediction algorithm is usable few days before the flight until the commencement of the flight to be able to produce accurate predictions.

In the case of turnaround duration prediction, it is appropriate to emphasize the usefulness of predictions which reveal whether a flight will be delayed, in other words, whether the actual turnaround time is greater than the available turnaround time. This is due to the fact that delay related costs occur in the case when actual turnaround time is greater than available turnaround time as in that case the flight will depart late from schedule. In situations when the actual turnaround time is predicted to be less than the available turnaround time, slack time is included in the turnaround schedule and there is more flexibility in terms of time for individual subprocesses to be completed.

Even though slack time would be included in the turnaround, it is not said that the departure flight will depart before scheduled time as passengers are required to arrive at the gate according to original schedule and flight is not allowed to leave ahead schedule if all passengers have not yet arrived (Eurocontrol, 2018). Also, other subprocesses might be completed only just in time with respect to the original departure schedule depending on their current capacity. According to these factors it is possible to draw a conclusion that it is more valuable to be able to make a prediction which indicates if a flight will depart late, compared to a prediction which shows that original scheduled departure time can be maintained. Even though a prediction of an on-time turnaround might also be valuable, it is still considered as the expected scenario which does not require additional measures to be taken. A prediction indicating a delay, on the other hand, is an exception to the original plan and requires actions to be taken to mitigate the delay. In the longer time span, during schedule planning when schedules are determined, reliable predictions of turnarounds with shorter than usual durations could be beneficial, but that is out of the scope of this thesis.

Based on data collected by Eurocontrol (2020), average arrival delay in Europe during the year 2019 was 12.2 minutes whereas average departure delay was 13.1 minutes. These figures are total delays and include all different causes for delay. 5.7 minutes of the total average departure delay are due to reactionary reasons and 3.4 minutes are due to airline operations (Eurocontrol, 2020). According to material from Eurocontrol (2018), reactionary delays refer to delays propagated from earlier flights due to late arrival of incoming aircraft or due to connecting crew or cargo. Delays related to airline operations, on the other hand, are caused by factors such as delays in boarding or baggage handling, aircraft cleaning, fuelling and catering, and technical defects (Eurocontrol, 2018). In this case reactionary reasons represent a fraction of 44% of the total delay and airline operations correspond a 26% fraction of the total delay. It can be concluded that delay causes under the airline operations category, which overlaps with turnaround activities, comprise one-fourth of all delay causes and can be considered to have a meaningful impact on the schedule adherence. Reactionary delays comprise almost half of all delay causes and the root cause for a reactionary delay can possibly be in some cases related to turnaround performance of a previous flight.

Data related to arrival and departure punctualities with different thresholds for flights in Europe during 2019 indicates that 59.4% of flights arrived before schedule or were on time whereas 40.6% of flights arrived late (Eurocontrol, 2020). This implies, that on average, a considerable part of flights arrives late and causes a reduction to their available turnaround time. Based on the same data, during 2019 only 15.9% of flights departed before scheduled time and 46.4% of flights departed late. In other words, departure punctuality figure reveals that only a small part of flights is able to depart before scheduled departure time and almost half of all flights are late from the original schedule. The distributions of arrival and departure delays are visualized in Figure 1.

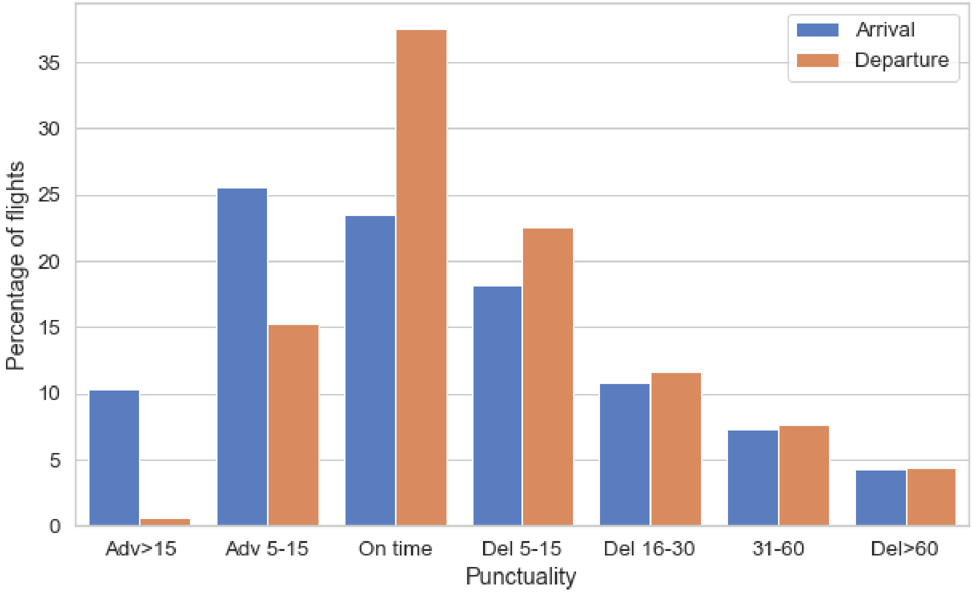


Figure 1: Arrival and depature punctuality 2019 (Eurocontrol, 2020)

Following findings from the 2019 delay data confirm the importance of turnaround time predictions: aircraft often arrive and depart late, early departures are rare, departure delay is affected by ground processes, and reactionary delays represent almost half of all delays which from operative perspective can be used in the prediction of subsequent flights.

In general, the information about delay is more valuable compared to information about schedule adherence as noted before. Normally turnarounds for short-haul aircraft vary between 30 to 55 minutes, depending on the scheduled buffer (Fricke & Schultz, 2009), and based on the information provided by Eurocontrol (2020), for European flights the average departure delay was 13,1 minutes during year 2019. From this information we can note that even for longer turnarounds (55+ minutes) the average departure delay corresponds to a notable fraction of the total turnaround length. According to research related to turnarounds (Gao, Huyan & Ju, 2015; Wan et al., 2019; Hassel, 2019), the mean absolute error range for turnaround predictions is between 4 to 10 minutes.

Existing research will be discussed later, but if the turnaround prediction accuracies mentioned above were used as benchmark values, it is possible to note that even though such predictions accuracies might not enable perfect optimization of turnaround actions, they might still give approximations about a turnaround duration and act as a support for operative decision making. In addition, predictions accompanied with contributing factors convey more information about the reasons behind possible long durations. Such information combined with reasoning that is based on domain expertise enables even better help for decisions about turnaround related strategies.

Predictions related to turnaround time and the information about the dominant factors behind a prediction might be usable for airline operations and the flight crew as well as the ground handling company. This is due to the fact that these stakeholders have the capability and resources to make effective decisions related to turnaround management. In the methodology part of the thesis, motivation for the machine learning based prediction algorithm will be discussed and XGBoost will be presented in more detail.

2.5 Previous research

Previous research related to aircraft turnarounds and its subprocesses exists from multiple viewpoints. According to San Antonio, Juan, Calvet, Casas and Guimarans (2017), in the scientific literature turnarounds have been analyzed in two different ways. Firstly, turnarounds can be viewed from task-based view where each subprocess is analyzed individually and its contribution to the total duration of the turnaround is assessed. In task-based view critical paths and bottlenecks in the turnaround process can be identified by analyzing single turnaround activities. Secondly, turnarounds can be viewed from a more holistic viewpoint in the aggregated approach where no details of different subprocesses are analyzed. Instead, in the aggregated approach the purpose of the analysis is to determine a scheduled turnaround time with sufficient buffer times based on the trade-off between possible delay costs and costs related to keeping the aircraft on ground more than necessary, according to San Antonio et al. (2017).

Oreschko et al. (2012) have predicted turnaround durations with stochastic process times. In their research, they produced a more accurate estimate of the turnaround duration than just “a best guess” by fitting distributions to empirical data and used a Monte Carlo Simulation to generate samples from the distributions to simulate the stochastic nature of turnaround subprocesses. In this approach a critical path is constructed by addition of sequential sub-process durations. Oreschko et al. (2012) provide four different prediction levels depending on the available information at the time of the prediction. Accuracy of the prediction gets better as the actual turnaround gets closer in time. This kind of study is an example of a task-based view to turnaround analysis. Similar to Oreschko et al. (2012), Wu and Caves (2004b) have used Monte Carlo Simulation combined with a Markovian type simulation model to analyze the relationship between departure punctuality and uncertainties related to a turnaround process.

Wang et al. (2017) have developed a turnaround time estimation model which is based on discrete time simulation. In their study, they have focused on the deboarding and boarding process as it is considered to account for a major part in terms of duration of a turnaround process. Durations for subprocesses, such as refuelling and cargo container loading, have been adopted from aircraft manuals whereas passenger behavior in boarding is simulated with an explicit function.

In addition to Monte Carlo Simulation and other modelling methods, machine learning algorithms have been applied both to turnaround and aircraft delay predictions. Chen and Li (2019) have used machine learning to model chained predictions of flight delays. From turnaround point of view, departure delay is strongly affected by the arrival delay of the previous flight leg and Chen and Li (2019) have used weather and air traffic performance data together with a random forest classifier for the modelling and prediction of delay propagation through a chain. Compared to task-based approach, this is an example of an aggregated approach to inspect air traffic delay prediction through available turnaround time without data related to individual subprocesses.

Horiguchi et al. (2017) have studied the use of random forests, XGBoost and neural networks in the prediction of flight delay. Furthermore, they applied these methods to create a binary classifier to predict if a flight is late or not. In their study, Horiguchi et al. (2017) provide different prediction accuracies depending on the time between the prediction and the actual flight. They also provide examples of actions that can be taken in order to minimize the possible delay such as better gate assignment, number of available ground handling crew or adjustment of duration of individual turnaround subprocesses. Horiguchi et al. (2017) have used information related to date of flight, departure and arrival airports, aircraft type as well as number of passenger and other passenger related attributes as features in their model.

Some other machine learning methods used in the prediction of aircraft flight delay are gradient boosted trees (Manna et al. 2017) and AdaBoost with k-nearest neighbors (Choi et al., 2017). Compared to flight delay prediction, notably less scientific literature is related to the use of machine learning methods in the prediction of aircraft turnarounds.

Choi et al. (2017) discuss an important topic related to misclassification costs that are asymmetric in classification of whether a flight is late or not. A false positive prediction, which indicates a delay even though if in reality there would not be circumstances leading to a delay, can cause additional procedures to take place for no actual reason. In reality however, a prediction can be confirmed as false only afterwards, assuming that the prediction algorithm has all the relevant information related to that specific scenario.

Gao, Huyan & Ju (2015) have used neural networks to estimate turnaround times and the features they have included in the model contain information about aircraft position (gate or stand), aircraft type, flight type (domestic or international), airline agent (ground handling company), arrival time, and the amount of passengers onboard. On the other hand, Wan et al. (2019) predict turnaround time by using support vector machines. In their model (Wan et al., 2019), the information about all the included features is not available but at least information about aircraft type, scheduled arrival and departure times as well as actual arrival and departure times are provided. The neural network model has a relative prediction accuracy of 25% for 85% of test samples and absolute error is 10min for nearly 78% of test samples. In the model based on support vector machine, the mean absolute error is 6.8

minutes. Both of the regression models provide point estimates for a turnaround duration.

Hassel (2019) has combined the critical path method and the use of machine learning methods by predicting cycle times, that is, subprocess durations, with a Random forest and a neural network. When lengths of individual subprocesses have been predicted, it is possible to determine the length of the turnaround by defining the critical path of subprocesses. In the thesis provided by Hassel (2019), similar features such as number of passengers or amount of fuel have been used compared to previous research. Such data has been supplemented with empirical time stamp data from start and end times of different subprocesses such as start and end of boarding or the time of closing the ground process by the dispatcher. This way Hassel (2019) has brought transparency to the prediction by providing the most influential subprocesses compared to standalone numerical duration predictions.

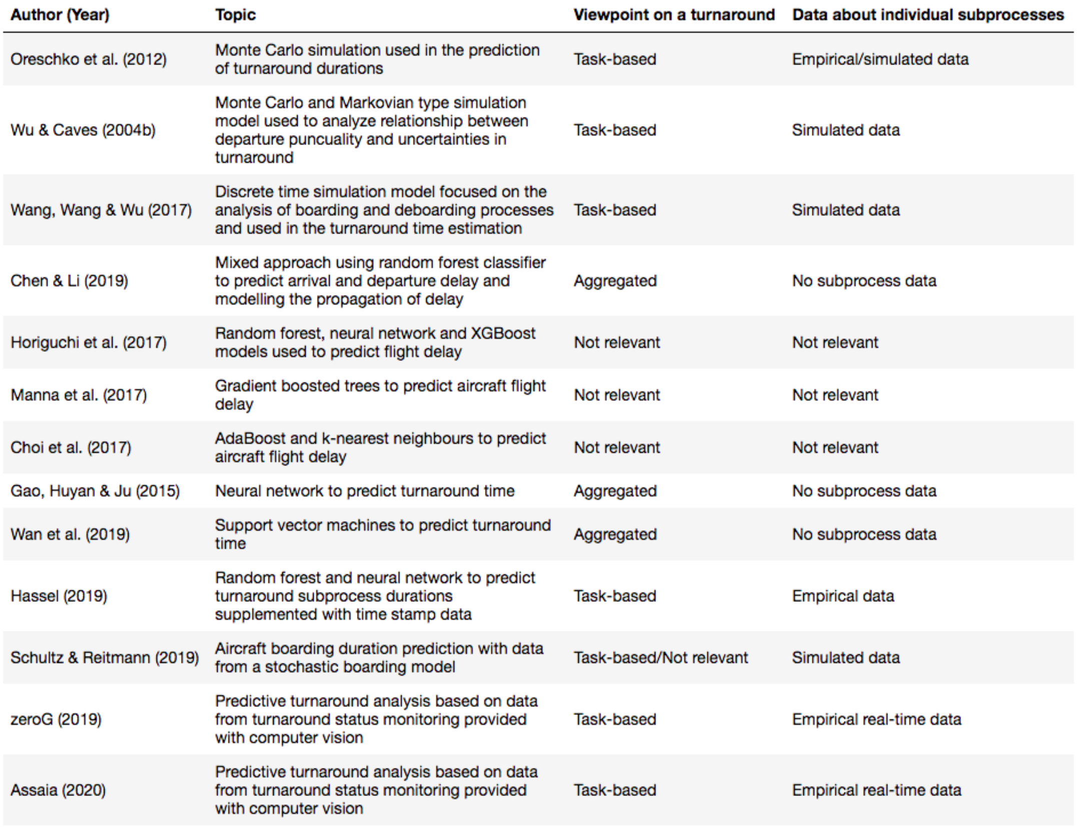
Article by Schultz and Reitmann (2019) is an example of a related article where machine learning has been used to predict aircraft boarding. However, in that study machine learning model is trained with data from a stochastic boarding model simulation as time stamps or actual data inside the plane is not available. This reflects the idea presented by Wu (2008) that due to lack of time stamps in many of the subprocesses in a turnaround, turnaround process can in many ways be considered as a black box.

Deep Turnaround (zeroG, n.d.) and Assaia Apron AI (Assaia, n.d.) both represent examples where machine vision is used to monitor and collect information about ongoing turnarounds. Information collected by these systems is further processed and predictive analysis is provided together with real-time information about the status of the turnaround (zeroG, n.d.; Assaia, n.d.). Such systems are examples where the lack of time stamps of turnaround subprocesses can be compensated by automatically generating process information received by the machine vision. From predictive modeling point of view, time span for the predictive information provided by such systems is short and not available before the turnaround.

Table 1 below contains a summary of the discussion regarding the previous research. Review of the previous research reveals that studies related to the prediction of turnaround duration with machine algorithms exist but opportunities for further contributions exist as well. Related studies (Gao et al., 2015; Wan et al., 2019) provide general models where input features such as aircraft type, passenger numbers, airline agent or stand information are provided. However, according to the stochastic nature of a turnaround process there are many additional factors that could provide explanation to the time that an aircraft requires for a turnaround. Such factors could be passenger type, amount of baggage or cargo, and the need for refuelling or de-icing. Similar features, such as passenger related attributes, have been used in the study conducted by Horiguchi et al. (2017) even though they are used for another purpose, in this case a binary classifier of aircraft delay. On the other hand, the lack of data has been tackled by Hassel (2019) who has augmented general turnaround data with empirical subprocess related time stamps. Hassel (2019) has combined task-based approach with aggregated approach and that way has been able to provide transparency to the turnaround process.

In addition to a more comprehensive set of different features, interpretability of the machine learning algorithm is useful from an operative point of view. Information about the factors that have resulted in the final prediction and their magnitude could provide the ability for airline or airport operators to make necessary changes that could minimize or even prevent the delay of an aircraft due to long turnaround times. Hassel (2019) has introduced transparency and interpretability to his model with the price of requiring more data and a more complex model. Gao et al. (2015) and Wan et al. (2019), in comparison, have addressed turnaround prediction as a black box and introduced a simpler model to predict a numerical duration of the turnaround. In general, it can be said that unlike the aggregated approach, task-based approach in turnaround predictions provides interpretability to certain extent by design.

Table 1: Summary of existing scientific literature



Author (Year)	Topic	Viewpoint on a turnaround	Data about individual subprocesses
Oreschko et al. (2012)	Monte Carlo simulation used in the prediction of turnaround durations	Task-based	Empirical/simulated data
Wu & Caves (2004b)	Monte Carlo and Markovian type simulation model used to analyze relationship between departure puncuality and uncertainties in turnaround	Task-based	Simulated data
Wang, Wang & Wu (2017)	Discrete time simulation model focused on the analysis of boarding and deboarding processes and used in the turnaround time estimation	Task-based	Simulated data
Chen & Li (2019)	Mixed approach using random forest classifier to predict arrival and departure delay and modelling the propagation of delay	Aggregated	No subprocess data
Horiguchi et al. (2017)	Random forest, neural network and XGBoost models used to predict flight delay	Not relevant	Not relevant
Manna et al. (2017)	Gradient boosted trees to predict aircraft flight delay	Not relevant	Not relevant
Choi et al. (2017)	AdaBoost and k-nearest neighbours to predict aircraft flight delay	Not relevant	Not relevant
Gao, Huyan & Ju (2015)	Neural network to predict turnaround time	Aggregated	No subprocess data
Wan et al. (2019)	Support vector machines to predict turnaround time	Aggregated	No subprocess data
Hassel (2019)	Random forest and neural network to predict turnaround subprocess durations supplemented with time stamp data	Task-based	Empirical data
Schultz & Reitmann (2019)	Aircraft boarding duration prediction with data from a stochastic boarding model	Task-based/Not relevant	Simulated data
zeroG (2019)	Predictive turnaround analysis based on data from turnaround status monitoring provided with computer vision	Task-based	Empirical real-time data
Assaia (2020)	Predictive turnaround analysis based on data from turnaround status monitoring provided with computer vision	Task-based	Empirical real-time data

This thesis addresses the turnaround duration estimation with an aggregated approach that considers turnaround process more as a black box with explicit input parameters similar to Gao et al. (2015) and Wan et al. (2019). In addition, the model in this thesis attempts to provide explanations and interpretability through additional explanatory models that utilize the trained machine learning model. With such a novel approach, this thesis makes contribution by providing a model that is more usable to situations when detailed time stamp data is not available but in the same time provides transparency to the predictions. It may even be able to give additional insights about the global effects of individual variables on the subprocesses and the overall turnaround time. Such approach can be described as explainable aggregated approach and compared to the other two approaches provided by San Antonio et al. (2017), explainable aggregated approach is a more developed version of the aggregated approach. The distinction between these three approaches is clarified in Table 2 below. The machine learning model XGBoost, which is used in this thesis together with explanation model SHAP, will be presented in the methodology section.

Table 2: Comparison of different approaches to turnaround analysis

	Task-based	**Explainable aggregated**	**Aggregated**
Required data	Detailed subprocess data	Aggregated input parameters	Aggregated input parameters
Provides interpretability	Yes	Yes	No
Contribution of subprocesses to the total duration	Provided through data	Inferred	Not known

3 Research objective and research questions

The turnaround process was introduced in section 2. Also, delays and on-time performance have been discussed with emphasis on the importance of predictability of a delay. Background information together with literature review have revealed the current state of research related to turnarounds as well as the opportunities for further scientific contributions.

In comparison to existing scientific literature around the topic, this thesis focuses on interpretable predictions that are based on the aggregated approach towards a turnaround, that is, considers turnaround process as a black box with certain input parameters resulting in an explainable prediction. As such, this study contributes to the scientific literature related to airline operations management by combining an existing machine learning algorithm with an interpretability method to provide operative insights in a novel manner.

Based on the discussion in the section 2 we can draw the objective for this thesis which is to create a proof-of-concept model for operative decision making which provides interpretable predictions for the duration of a turnaround. In more detail, aim is to provide interpretability on two levels: on a local level which is the interpretability of individual turnarounds and its contributing factors as well as on a global level which refers to general patterns in turnaround durations in the airport in question.

In the case of a black box approach on the turnaround duration prediction, the question is eventually related to interpretability versus data need. More data about different factors may provide better interpretability but what are the factors that are enough to create an interpretable predictive model? Based on the research objective, focus of this thesis will be guided by following research questions:

1. Which turnaround related parameters are needed to predict turnaround durations?
2. How are input parameters of a turnaround capable of explaining the turnaround process and provide interpretability?

Assumption is that due to the nature of machine learning methods, some number of features and data are able to predict turnaround durations without explicitly modelling the subprocesses. In addition, explanatory models together with the machine learning algorithm can increase interpretability and provide operationally useful results. In the methodology section, machine learning and interpretability will be introduced to further extent.

4 Methodology

This section describes the used methods for this research as well as the dataset describing the turnaround process. Firstly, the research approach and nature of this study is discussed followed by the introduction of the concept of machine learning. Secondly, a more detailed introduction to the XGBoost model is provided with the introduction to the interpretability framework SHAP. Finally, the collected data together with the essential analysis is provided.

4.1 Research approach

According to Wohlin, Höst and Henningsson (2003), in a quantitative research the aim is to identify a causal relationship between different factors. In this thesis the research is conducted in a quantitative manner to identify a possible connection between a certain set of variables in an aircraft turnaround and the duration of a turnaround. More precisely, the aim of this thesis is to provide interpretability for future turnaround duration predictions which is derived from the relationship between a given a set of independent variables, such as number of passengers or amount of fuel need, and the dependent variable, that is the turnaround duration.

A case study research is a research project where data is collected from a real project for a specific purpose and that purpose guides the research throughout the process (Wohlin et al., 2003). This thesis follows the principles of a case study by inspecting history data from a certain airport and aims to create a statistical proof-of-concept model that is able to produce predictions about future scenarios as well as able to provide interpretability for these predictions.

In their book, Kuhn and Johnson (2013), provide general set of steps to a machine learning problem which are data gathering and preparation, feature selection, model selection and hyperparameter tuning as well as the measurement of model performance. These steps form the foundation for the approach to the part of the machine learning modelling in this thesis. On the other hand, the means to provide interpretability for predictions are partly adopted from the steps and methods discussed by Molnar (2019) in his book "Interpretable Machine Learning".

4.2 Machine learning and supervised learning

Machine learning consists of a set of methods which are used to find patterns from data automatically (Murphy, 2012). According to Bzdok, Altman and Krzywinski (2018), these discovered patterns can be used to create predictions about unseen data and forecast future behavior. They (Bzdok et al., 2018) point out that such forecasts can help in the identification of subsequent actions without the requirement to fully understand the underlying mechanisms behind the data. If this idea is applied to aircraft turnaround, it is possible to assume that the general principles and causalities between turnaround subprocesses are known but with certain combinations of turnaround related parameters some new patterns might arise.

Supervised learning is a technique of machine learning where data is provided for a learning algorithm and the goal is to learn a mapping $f(\mathrm{x})$ which creates output y for each input x. This technique assumes a labeled set of training data $\mathcal{D} = \{(\mathrm{x_i}, \mathrm{y_i})\}_{\mathrm{i}=1}^{\mathrm{N}}$ (Friedman, Hastie & Tibshirani, 2001). This can be considered as a function approximation problem where we train our model to eventually approximate the unknown function which in this case corresponds to the turnaround process duration. In other words, given the input parameters such as number of passengers or fuelling need, we get the duration as an output. Important property of the supervised learning technique is that we can adjust, or train, our approximation model based on the difference $y_i - \hat{y}_i$ which is the difference between output value from training data and the approximation generated by the model (Friedman et al., 2001). Based on Vapnik (1999), the difference between the output value based on the training and the output given by the model is called the loss and it acts as a metric for the performance of the model. The loss function, which can have many different forms, is calculated for certain amount of inputs and the sum of those inputs is minimized (Vapnik, 1999).

The goal of a supervised learning algorithm is to learn the mapping with a required accuracy from the training data and be able to make predictions based on novel data (Murphy, 2012). Different types of models to approximate the unknown function exist such as decision forests or neural networks (Jordan & Mitchell, 2015). In this study XGBoost is used to map input values to output values and approximate the unknown function f.

In this thesis the approach towards the turnaround duration prediction through a machine learning algorithm compared to other statistical methods is justified by two ideas. Firstly, the purpose of this thesis is to create an interpretable model that is able to predict turnaround durations in different settings. Data provided in this study is from a single airport and the aim is to create a proof-of-concept model that can be scaled to other scenarios. This is backed by the idea from Bzdok et al. (2018) that machine learning has a focus on general-purpose learning algorithms to find patterns from data whereas more traditional statistical methods have focus on the creation and fitting of probability models on the data which is project specific. From statistical perspective, multiple probability models might be needed for different airports and aircrafts. On contrary, from supervised learning aspect one model might be enough to cover multiple cases and recognize differences between airport and aircraft combinations without the explicit fitting of distributions and recognition of critical paths.

Secondly, in many cases there is lack of data from the turnaround subprocesses as discussed in the previous section. In such situations the turnaround process needs to be considered as an aggregated black box model. Supervised learning models might find patterns from turnaround data without explicitly modelling the subprocesses and provide predictions based only on certain set of input parameters. According to Choi et al. (2017), delay prediction is hard to solve analytically but instead machine learning can find hidden patterns in the data. That same idea can be applied to turnaround duration prediction.

4.3 Gradient boosting and XGBoost

XGBoost (eXtreme Gradient Boosting) is a scalable tree boosting system introduced by Chen and Guestrin (2016). They specify that the implementation of XGBoost is based on the gradient boosting algorithm which is an ensemble learning method. Boosting refers to a technique where multiple sequential models are built, and each model attempts to improve the accuracy of the previous model, according to Mitchell & Frank (2017). XGBoost is a supervised learning algorithm and can be used for both classification and regression (Mitchell & Frank, 2017). In gradient boosting, as well as in XGBoost, the base model is a decision tree. By combining weak base learning models, such as decision trees, a better accuracy and robustness of the model is achieved (Friedman et al., 2001). In this thesis XGBoost is used for regression as the aim is to provide a continuous value and therefore the base model can be called a regression tree instead of a decision tree.

According to Chen and Guestrin (2016), XGBoost has been successful in many machine learning competitions especially against popular methods, such as neural networks, which have been known for their good performance. Important features of XGBoost are a good predictive performance due to its nature as an ensemble gradient tree boosting system, computational efficiency compared to traditional gradient boosting, ability to handle sparse data and the option for multicore computation (Chen & Guestrin, 2016). Ability to handle sparse data applies to both tree building phase and to predictions that are based on new observations. XGBoost is called a tree boosting system as it covers concepts such as column block for parallel learning, cache-aware access and blocks for out-of-core computation which are related to the efficient use of hardware (Chen & Guestrin, 2016). Therefore, XGBoost is not purely a statistical technique.

Friedman et al. (2001) point out that machine learning methods based on decision trees are a good off-the-shelf procedure for predictive learning. They specify that this is due to their interpretability, natural handling of discrete and continuous data, they are not affected by predictor outliers, they are insensitive to monotone transformations of the inputs and they are resistant to irrelevant predictor variables. A disadvantage of a decision tree is its inaccuracy, but gradient boosted models are successful in mitigating this issue (Friedman et al., 2001).

Many methods exist for supervised learning where the goal is to create predictions based on given data. Such predictive learning methods are differently suited for different occasions. However, it is hard to say which one is best suited for a given situation in advance according to Friedman et al. (2001). Based on the track record of successful applications and other factors discussed above, XGBoost was chosen as the model to approximate turnaround duration function in this thesis. As mentioned by Chen and Guestrin (2016), XGBoost has predictive performance that is comparable to neural networks.

Next, the concepts bias-variance tradeoff and regularization as well as their connection to XGBoost will be discussed. In addition, main principles of the XGBoost tree building algorithm will be presented followed by a discussion about hyperparameters. Full derivation of the related formulas required for XGBoost algorithm will be presented in appendix A.

4.3.1 Bias-variance tradeoff and regularization

Bias and variance are related to the generalization performance of a machine learning model (Friedman et al., 2001). As mentioned, the goal of supervised machine learning algorithm is to learn a mapping from training data and to be able to make predictions on new unseen data based on that learned mapping. This ability of a machine learning algorithm to make accurate predictions on novel data is called generalization.

According to Kuhn and Johnson (2013), complex models tend to overfit on training data which means that they have high variance. Variance is the amount of change in the function approximation if a different training set was used and high variance leads to poor generalization (James, Witten, Hastie & Tibshirani, 2013). On the other hand, simpler models have a tendency to underfit on the training data as they may not have enough flexibility to produce a correct mapping based on the training data. This concept is called bias and higher bias eventually also leads to poor predictive performance (Kuhn & Johnson, 2013).

Friedman et al. (2001) present that bias-variance tradeoff is a concept which means that machine learning model with high variance but low bias is complex and has a good accuracy on the training data but has a poor performance on new unseen data as the accuracy fluctuates even due to small changing details in the unseen test data. Models with high bias and low variance, on the other hand, are robust to details but may not have the ability to model relationships from data to a satisfying extent (Friedman et al., 2001). Taken to extreme, both situations result in poor generalization accuracy and the goal is to find a tradeoff between these two different errors for a machine learning model so that it performs well with both training and testing data.

In many cases complex machine learning models are required to provide sufficient predictive capability. Complexity of data and complex models combined with regularization techniques enables models to be trained on data sets which might have a limited size but where overfitting can be prevented (Bishop, 2006). XGBoost can be considered as a complex machine learning model and the regularization features used to prevent overfitting will be presented in the next section together with the XGBoost algorithm.

4.3.2 XGBoost algorithm

Objective function in Equation 1, is the core of the XGBoost algorithm (Chen & Guestrin, 2016; Zhang et al., 2018) and is presented as follows

$$\mathcal{L}^{(t)} = \sum_{i=1}^{n} l(y_i, \hat{y}_i^{(t-1)} + f_t(\mathbf{x}_i)) + \Omega(f_t) \qquad (1)$$

where $\Omega(f) = \gamma T + \frac{1}{2}\lambda \parallel w \parallel^2$.

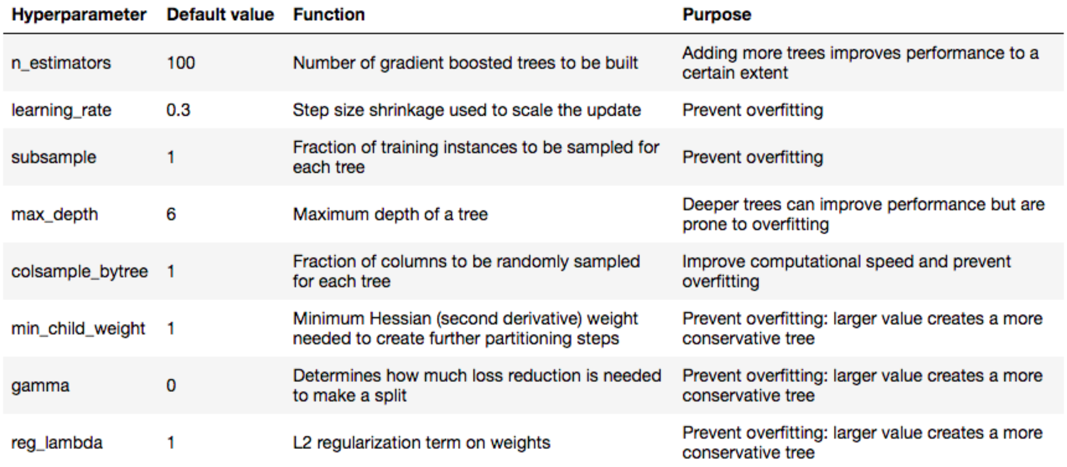
Objective function includes the loss function l and the regularization term $\Omega(f_t)$ which is used to prevent overfitting and also to provide support for arbitrary differentiable loss functions (Mitchell & Frank, 2017). Requirement for the loss function l is that it needs to be convex and differentiable. If the regularization term is set to zero in the objective function, it corresponds to the objective function in gradient tree boosting (Chen & Guestrin, 2016). For the loss function y_i corresponds to the value of the i-th target variable from training data whereas $\hat{y}_i^{(t)}$ is the predicted value for i-th instance at the t-th iteration. $f_t(\mathbf{x}_i)$ corresponds to the tree structure at t-th iteration, given i-th input instance. Therefore $l(y_{i,}\hat{y}_i^{(t-1)} + f_t(\mathbf{x}_i))$ measures the difference between target value from training data and the output given from the chain of tree structures until iteration $t - 1$ added to the output of the most recent tree.

Regularization term $\Omega(f)$ contains term γT where T is the number of leaves in the tree and the term γT penalizes additional leaves. Term $\frac{1}{2}\lambda \parallel w \parallel^2$ penalizes extreme weights and resembles the L2 regularization. Term w refers to the predicted values at each leaf node, that is, the leaf weights. Terms γ and λ are configured by the user. In this thesis, the default loss function squared error loss will be used.

The building of a XGBoost structure starts with an initial prediction which is by default 0.5 and that is followed by the calculation of residuals for the rows in the dataset. Followed by that, XGBoost enumerates splits and picks a split for the tree node which maximizes gain (Mitchell & Frank, 2017). Based on Dhaliwal, Nahid & Abbas (2018), XGBoost continues building a single tree with given constraints, such as maximum tree depth, and when a tree has reached maximum depth, XGBoost prunes the tree based on gain values and γ. Eventually XGBoost calculates the output for a tree and adds it to the previous prediction factored by the learning rate. This process is continued until the given number of trees have been built (Dhaliwal et al., 2018).

As mentioned, there is a variety of parameters in XGBoost such as learning rate, column subsampling and row subsampling which are used to prevent overfitting, but which also need to be tuned according to the dataset. Table 3 below presents all the hyperparameters of XGBoost which will be tuned in this study to adjust the algorithm for turnaround duration predictions. Information in the table is collected from Xgboost developers (2020).

Table 3: Descriptions for hyperparameters of XGBoost



Hyperparameter	Default value	Function	Purpose
n_estimators	100	Number of gradient boosted trees to be built	Adding more trees improves performance to a certain extent
learning_rate	0.3	Step size shrinkage used to scale the update	Prevent overfitting
subsample	1	Fraction of training instances to be sampled for each tree	Prevent overfitting
max_depth	6	Maximum depth of a tree	Deeper trees can improve performance but are prone to overfitting
colsample_bytree	1	Fraction of columns to be randomly sampled for each tree	Improve computational speed and prevent overfitting
min_child_weight	1	Minimum Hessian (second derivative) weight needed to create further partitioning steps	Prevent overfitting: larger value creates a more conservative tree
gamma	0	Determines how much loss reduction is needed to make a split	Prevent overfitting: larger value creates a more conservative tree
reg_lambda	1	L2 regularization term on weights	Prevent overfitting: larger value creates a more conservative tree

4.4 Model interpretability

In many cases the information of why a predictive machine learning model ends up in a certain outcome may be as important as the accuracy of the prediction (Lundberg and Lee, 2017). According to Doshi-Velez and Kim (2017), this is especially the case when predictions created by a machine learning model are used in operative decisions and there are juridical, monetary or safety-related consequences for actions based on faulty reasoning by the model. Doshi-Velez and Kim (2017) emphasize, that the need for interpretability comes from the incompleteness of the problem formalization. In the case of an aircraft turnaround duration prediction this means that the only problem is not to create an accurate estimate but in addition to be able to reveal reasons for a certain prediction and to potentially discover patterns that the machine learning model has identified. Interpretability can therefore reveal valuable insights from the data.

To put this into another perspective, with interpretability it may be possible to spot anomalies and recognize erroneous predictions when the reasons behind the predictions are known. One notorious example of erroneous reasoning by the machine learning model can be found from the domain of image classification where a model separated images of huskies from the images of wolves not based on the attributes of the animal but based on the snowy background (Ribeiro, Singh and Guestrin, 2016).

According to Doshi-Velez and Kim (2017), interpretability in machine learning domain means the ability to explain outcomes in a format that is understandable to humans. Moreover, Gilpin et al. (2018) state that an explanation of a machine learning model is good when you can no longer keep asking why a model has resulted in a certain outcome. In his book, Molnar (2019) uses terms interpretable and explainable interchangeably and that convention is also followed in this thesis.

4.5 Interpretability methods and their limitations

There exist multiple ways to provide interpretability for machine learning models. One way to obtain interpretability is to have a simple model where the model itself can be analyzed (Bouneder, Léo and Lachapelle, 2020). This method works for models such as decision trees and linear regression which are self-explanatory as themselves due to their simplicity. Based on this idea Molnar (2019) divides interpretability into intrinsic and post hoc methods. Intrinsic methods provide interpretability by restricting the complexity of the model whereas post hoc methods are applied after the machine learning model has been trained. Complex models such as XGBoost are more accurate than simpler models but need to be interpreted by post hoc methods.

Following the taxonomy presented by Molnar (2019), interpretability can be also divided into model-specific and model-agnostic as well as into local and global methods. Model-specific methods are designed only for certain machine learning models. Model-agnostic methods, on the other hand, can be used on any machine learning model and such methods typically make inferences by analyzing varying feature input and output pairs. Global interpretability methods reveal general patterns that explain behavior of the entire model whereas local methods are used to explain individual predictions. Doshi-Velez and Kim (2017) state that global methods can be used to make generalizations of the data and local methods can be used to justify specific decisions based on the predictions given by the machine learning model.

Main advantage related to model-agnostic interpretation methods is their flexibility (Molnar, 2019). In practice this means that a model-agnostic interpretation model is independent of the underlying machine learning algorithm and can therefore be used to compare interpretability between different models. However, Lundberg et al. (2020) note that model-agnostic interpretation models can be slow and provide variable results especially for local explanations. Examples of tools for model-agnostic interpretations are partial dependence plots (PDP), individual conditional expectation (ICE) plots, local interpretable model-agnostic explanations (LIME) and Shapley additive explanations (SHAP) which are based on the Shapley values (Molnar, 2019).

Molnar (2019) presents that the idea of partial dependence is that it calculates how the average prediction of the machine learning model changes when a certain feature is changed. In other words, the predictions are marginalized over other features resulting in a function that gives predictive value based only on the feature that is being inspected. The disadvantage of partial dependence is that it does not take correlation between features into account (Molnar, 2019). Individual conditional expectation is similar to the concept of partial dependence but instead it calculates the relationship between the prediction of the model and the feature for individual observations (Goldstein, Kapelner, Bleich & Pitkin, 2015). Both of these methods can be utilized in plots and therefore they act mainly as a visual support. As individual conditional expectation plots show individual lines for single observations, it can uncover possible interaction effects (Goldstein et al., 2015).

According to Ribeiro et al. (2016), local interpretable model-agnostic explanations, or LIME, is a local surrogate model that inspects the interpretation of a single instance in a dataset. LIME creates a new dataset by making variations to the input data and monitoring changes in the machine learning model output. They explain (Ribeiro et al., 2016) that after the creation of the new dataset, LIME assigns weights to the rows in the new dataset based on their proximity to the instance that is being inspected. Finally, after the creation of the dataset with predictions and permutations from the original data with weights, LIME creates a new local interpretable model based on the new dataset and provides interpretability through that model (Ribeiro et al., 2016). Molnar (2019) notes that LIME works for many different kinds of data but especially for tabular data the weakness of LIME is the difficulty to determine proximity, or neighborhood, for the inspected instance. In addition, Molnar points out the possible instability of explanations in LIME.

Compared to model-agnostic interpretation models, model-specific tools such as Tree SHAP can be faster to compute (Lundberg & Lee, 2017). Disadvantage of model-specific interpretation models is that they are bound to a certain machine learning model (Molnar, 2019). Lundberg, Erion and Lee (2018) present that many tree ensemble machine learning models such as XGBoost, gradient boosting and random forests are equipped with built-in feature importance methods where the purpose is to find which features drive the machine learning predictions the most.

These model-specific feature importance values are global values which are calculated for the whole dataset and established for each feature, or column, of the dataset, according to Lundberg et al. (2018). In these methods feature importance can be calculated based on the gain, split count and permutation. For a certain feature, gain is the total reduction of impurity achieved when that specific feature is used as a split for all possible splits. On the other hand, split count refers to the number of cases where the feature is used as a split in the final model. In the calculation of feature importance based on permutation, values of a feature are randomly permuted in the test set followed by the observation of change in the model's error. Lundberg et al. (2018) point out that in this type of feature importance, the relationship between the feature and the true output of the machine learning model is broken and therefore a single feature is more valuable the more the error increases by permuting that specific feature.

Even though these different feature importance types above are specific to tree ensemble models and their implementations are provided by their algorithm implementations, especially permutation feature importance in general can be considered as a model-agnostic interpretability method, according to Molnar (2019).

Lundberg et al. (2018) note that feature attribution methods where feature importance is calculated either with gain, split count and permutation are inconsistent. This can be manifested through occasions when a machine learning model is changed so that a feature has a higher impact on the output as before but the importance for that specific feature now has a lower value. In practice, Lundberg et al. (2018) discuss, this means that a feature with high attribution towards the output may have a lower importance value than a feature that in reality has a lower impact on the models output and therefore comparison of importance values between features might lead to false assumptions.

Compared to feature attribution methods discussed above, SHAP as a method to calculate feature importance values is consistent, based on Lundberg et al. (2018). SHAP values itself are part of the model-agnostic interpretation methods, meaning that they can be used for any machine learning model, but the more efficient derivation called Tree SHAP is a model-specific interpretation method designed for tree ensemble machine learning models (Lundberg et al., 2018). The library for calculation of SHAP values also enables the calculation for interaction effects between features (Lundberg et al., 2020).

Table 4 below contains a comparative summary of different interpretation methods. According to Hall (2019), it might be useful to use several techniques to explain and provide interpretability for machine learning models. Due to the versatile features of the Tree SHAP, it will be used together with partial dependence plots to inspect the turnaround process and give insights into the predictions. Turnaround process will be examined both in the global and local level of explanations. In the next section the calculation the Shapley values will be presented together with details about SHAP and Tree SHAP.

Table 4: Comparison of interpretation methods

Model-agnostic methods - Flexible: available for any ML method - Provides comparable results between ML models - Can be slow and produce variable results on local level	**Model-specific methods** - Available only for certain models - Bound to certain model: hard to compare results between different models - Potentially faster to compute
Partial dependence plots - Visual tool	Feature importance based on gain and split count - Native methods - May be inconsistent
Individual conditional expectation plot - Visual tool	Tree SHAP - Based on Shapley values - Consistent - Faster calculation compared to SHAP
LIME - Potential instability of explanations	
SHAP - Based on Shapley values - Consistent	

4.5.1 Shapley values, SHAP and Tree SHAP

SHAP is a framework that is based on the Shapley values (Lundberg & Lee, 2017). Originally Shapley values were used in the domain of game theory for determining the worth of each player over all possible combinations of players in a multiplayer co-operative game (Lipovetsky & Conklin, 2001). In the case of predictions given by a machine learning model, features can be considered as players and co-operative game is the prediction of a particular instance. According to Molnar (2019), Shapley value of a certain feature is its contribution to the prediction. That contribution is weighed and summed over all possible combinations of feature values.

Equation 2 and Equation 3 below are presented by Molnar (2019) but are based on the idea of Shapley sampling values by Lundberg and Lee (2017). In the original calculation of Shapley regression values for linear models, the model is trained for all the different subsets of features. According to Lundberg & Lee (2017), in the Shapley sampling values calculation, on the other hand, the effect of removing a feature from the model is approximated by integrating over samples from the dataset. In addition, a method called sampling approximation for the different models is performed. This way Shapley sampling values avoids the need to train the model multiple times and also allows a smaller number of differences to be computed (Lundberg

& Lee, 2017). Formulas for the calculation of Shapley values are determined so that

$$\phi_j(val) = \sum_{S\subseteq\{x_1,\dots,x_p\}\setminus\{x_j\}} \frac{|S|!(p-|S|-1)!}{p!}\left(val\left(S \cup \{x_j\}\right) - val(S)\right) \tag{2}$$

where $val_x(S) = \int \hat{f}(x_1,\dots,x_p)d\mathbb{P}_{x\notin S} - E_X(\hat{f}(X))$. (3)

Equation 2 is the calculation of the Shapley value for a certain feature. S corresponds to the subset of features used in the model and therefore |S| is the size of that subset. p is the number of features and x corresponds to the vector of feature values of the instance that is explained. Equation 3 consists of the prediction of feature values that belong to set S followed by the marginalization over features that are not present in that set. $E_X(\hat{f}(X))$ is the expected value of the prediction, or the so-called baseline value, as presented by Molnar (2019).

Shapley values provided by the SHAP framework are called SHAP values. Their unit is the unit of the prediction space, in this case minutes, and the contribution of a feature presented by SHAP values can be either a positive or a negative value (Lundberg et al., 2018). In practice, SHAP can determine how much, for example deicing or the number of passengers, has effect on the duration of a single aircraft turnaround and also provide global importance values for different features.

Different ways exist to estimate Shapley values (Hall, 2018). Tree SHAP, or TreeExplainer, provided by the SHAP explanation model, is a specific implementation of Shapley values for tree ensemble machine learning algorithms. It is based on the traversal through the internal tree structure to estimate the impact of different features. In general, the computation of Shapley values is a NP-hard problem but for tree-based models exact Shapley values can be computed in polynomial time (Lundberg et al., 2020).

Molnar (2019) points out that largest advantage of SHAP is the strong theoretical foundation in the calculation of Shapley values. However, Molnar (2019) also notes that especially in Tree SHAP, features may have a SHAP value different from zero even if they do not have an effect on the predictions. This is due to the nature of Tree SHAP as it relies on the calculation of conditional expected function, according to Molnar (2019).

4.5.2 Global feature attribution and interaction effects in SHAP

In addition to the calculation of SHAP values and the provision of local explanations, SHAP explanation model provides global methods for interpretation of a machine learning model by aggregating SHAP values (Molnar, 2019). When local explanations are combined across the whole dataset, the resulting mean magnitude of SHAP values for a single feature can be presented as global feature importance values (Lundberg et al., 2019).

Lundberg et al. (2019) discuss that interaction effects can be distinguished from the main effects in SHAP. They present that by following the ideas from the calculation of Shapley values and the concept of Shapley interaction index, Tree SHAP can also reveal SHAP interaction values for explanations. In SHAP, interaction values comprise a matrix where main effects are on the diagonal and interaction effects on the off-diagonal. Main effect for a feature is the difference between the SHAP value and interaction effects between other features.

Global attribution effects and local explanations for a single prediction combined with possible interaction effects are properties of the Tree SHAP that can be used to interpret predictions given by the XGBoost from many perspectives. They aid in the interpretations of single predictions but also might give valuable insights related to the turnaround dataset in general. In the next section the dataset will be presented together with methods related to the preprocessing of the data.

4.6 Data

Dataset used in this study comprises of data collected from an airport between a time span of several years. From here on in this study, the airport will be referred to as airport A. Airport A was selected as the airport of interest for three reasons. First of all, airport A is one of the airports with the largest amount of daily operations from Helsinki meaning that it has the best potential to respond for the data need regarding the machine learning algorithm XGBoost.

Secondly, airport A provides simplicity compared to larger airports. According to Lohatepanont and Barnhart (2004), a hub-and-spoke network is a flight network structure which consists of a hub airport (Helsinki) and spoke airports (in this case, airport A). A spoke airport has mostly flights to and from the hub airport. This means that passengers arriving to airport A from Helsinki are most probably arriving at their final destination and passengers departing from airport A do not have previous connecting flights. Therefore, there are less factors affecting the turnaround as there is no need to wait for connecting passengers, baggage or cargo from other flights.

Thirdly, airport A seldomly has multiple simultaneous arrivals or departures by different airlines nor it is a congested airport. This means that the traffic situation and the interference due to simultaneous capacity demand by different turnarounds is marginal. In practice this means that availability of ground personnel and equipment is not usually compromised by simultaneous demand.

Based on the reasons above, it is possible to justify turnaround operations in airport A as the basis for the dataset. In addition, due to the simplicity of the airport A, it might be easier to reveal patterns and causalities between turnaround time and variables in data compared to larger and more complex airports where all the explaining factors might not emerge through available data features.

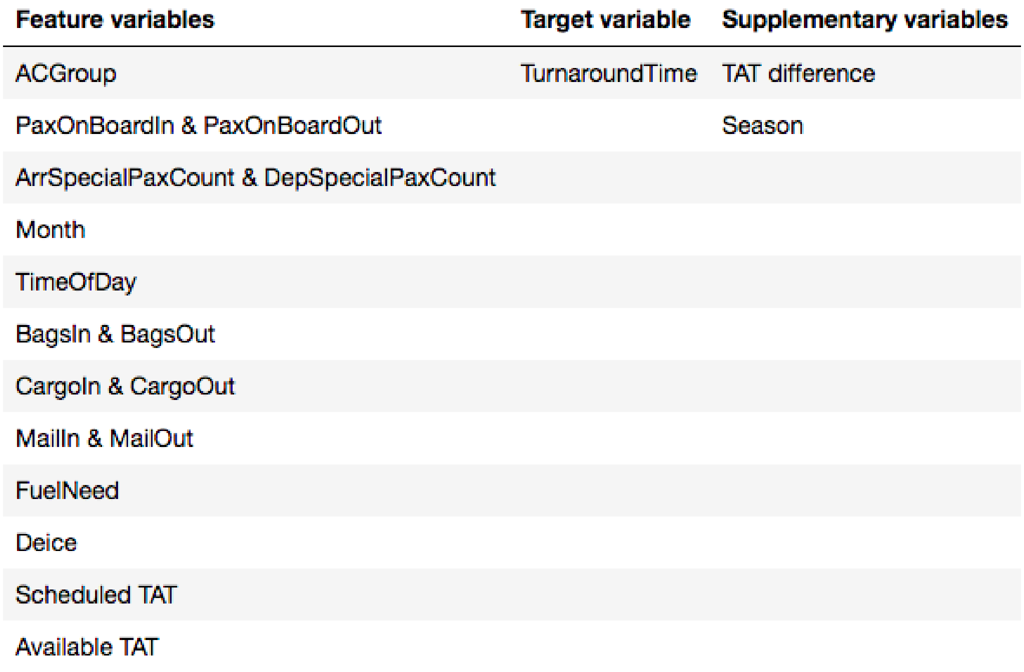
It is good to note that there exists other special characteristics related to airport A. Due to its proximity to Helsinki airport, most of the turnarounds in airport A do not include fuelling. Instead of fuelling at airport A, most flights to airport A from Helsinki perform tankering where fuelling at Helsinki covers also the fuel need for the return flight from airport A. For turnarounds in airport A this means that there is one subprocess less to be performed. In airport A nearly all turnarounds are performed on gate positions which means that there is no need to distinguish turnarounds performed on gate positions from turnarounds performed at remote stands.

Another special characteristic of airport A is that departure of flights which are returning back to Helsinki may be restricted by the traffic situation in Helsinki. During rush-hours in Helsinki, departure flights from spoke airports may be delayed in order to avoid unnecessary holding in the airspace of Helsinki airport. For turnarounds such delays result in situations where all the necessary subprocesses of a turnaround might be completed but the aircraft is unable to depart and the turnaround time is prolonged. This is an issue that a machine learning algorithm might be able to learn from a dataset. However, if such delays do not occur regularly for certain times of a day and if they have a random nature, it is possible to filter and remove such turnarounds from the dataset based on the delay code given for such flight.

4.6.1 Variables

Variables used in the analysis of this study are listed in Table 5 below. Feature variables are used as input features for the machine learning algorithm XGBoost to predict the duration of an aircraft turnaround in airport A. Features are selected based on domain expertise, in other words, based on the assessment of the usefulness of an individual feature and also based on the availability of data. In addition to input features, Table 5 contains the target variable TurnaroundTime and supplementary variables turnaround time difference (TAT difference) and Season. TAT difference indicates the difference between scheduled turnaround time and actual turnaround time and likewise the Season variable, TAT difference is not used for the final model but as an additional variable aiding in the exploratory data analysis.

Table 5: Different variables used in the analysis



Feature variables	Target variable	Supplementary variables
ACGroup	TurnaroundTime	TAT difference
PaxOnBoardIn & PaxOnBoardOut		Season
ArrSpecialPaxCount & DepSpecialPaxCount		
Month		
TimeOfDay		
BagsIn & BagsOut		
CargoIn & CargoOut		
MailIn & MailOut		
FuelNeed		
Deice		
Scheduled TAT		
Available TAT		

Issues such as passenger load factor (percentage of seats sold for a flight) and amount of cargo carried are major determinants of the profitability for a route (Antoniou, 1992). From a competitive perspective between airlines, such information can be considered as confidential. Therefore, visualization of data related to number of passengers and amount of cargo will be presented in this thesis in such a manner that main relationships between variables are conveyed while simultaneously the sensitiveness of information will be preserved. Unfortunately, in some occasions this will lead to non-optimal and less informative visualizations. Next, explanations for individual variables in terms of their relation to the turnaround process, their unit and a visualization of the distribution of values are presented.

ACGroup is the group of the aircraft for which the turnaround is performed, in this case a variant of the Airbus A320 family. Different variants of the A320 family include A319, A320, A321 legacy and the A321 sharklet models. From a turnaround point of view main differences between these models are passenger, cargo and fuel capacities (Airbus, 2020). A321 legacy and A321 sharklet versions do not have significant differences regarding said capacities and can be considered as one A321 group in this study.

In addition to differing passenger, cargo and fuel capacities between A319, A320 and A321, smaller A319 and A320 planes have a different fuel tank system compared to A321. A319 and A320 planes have small fuel tanks in their wingtips which during cold ambient temperatures and cold fuel temperatures may create frost over the wings. From a turnaround point of view this might be operationally significant as this accumulation of frost can be prevented by performing a fueling in Helsinki in which the fuel in the wingtips will be emptied before arriving at the destination airport A. However, this in many cases leads to a new refuelling need in airport A and adds a new subprocess to the turnaround. Another option would be to perform tankering which would remove the need to refuel in airport A but might lead to a need for deicing which again would an additional but different subprocess to the turnaround in airport A.

Division of flights between different aircraft is shown in Figure 3. For aircraft variant A319, deicing was performed for 22% of flights whereas for A320 the proportion is 21% and for A321 it is 14%, respectively. A319 presents 31% of all flights, A320 42% and A321 27%.

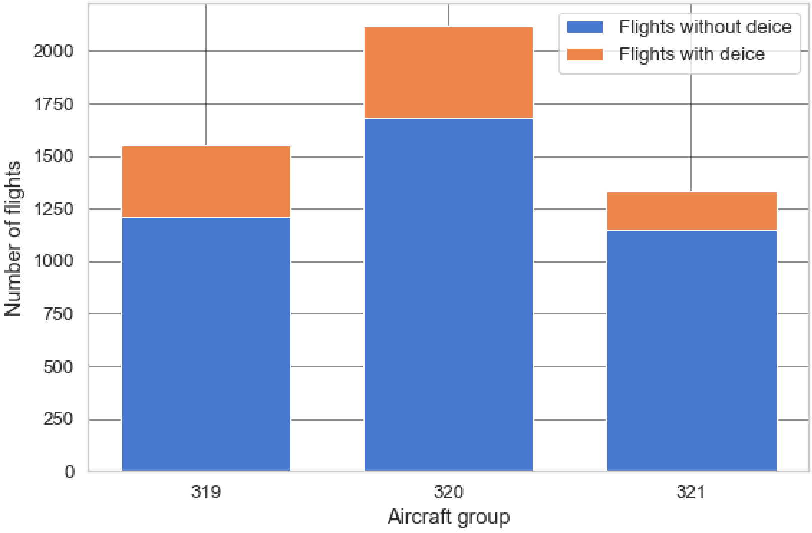


Figure 3: Division of flights between different aircraft variants and coloring based on whether deicing was performed or not

Features *PaxOnBoardIn* and *PaxOnBoardOut* refer to the number of passengers arriving and departing from airport A. In this study they will be treated as separate features as they are related to different processes of the turnaround, namely, the deboarding and boarding. As discussed in the literature review, passenger boarding is usually part of the critical path in a turnaround process. This means that the duration of boarding, which is dependent on the passenger count, is usually directly related to the turnaround duration. No qualitative information about passengers such as gender or age are passed on for the machine learning algorithm in addition to information regarding special passengers. Passenger capacities for the A320 family aircraft are a maximum of 144 passengers for A319, 174 passengers for A320 and 209 passengers for A321 versions. However, the actual capacities might differ based on the division between economy and business classes.

Features *ArrSpecialPaxCount* and *DepSpecialPaxCount,* on the other hand, refer to the number of arriving and departing special passengers. A passenger belonging to a special class may have reduced mobility of some sort which means that they might need aid in order to board and deboard the plane. Such passengers are usually boarded before other passengers and in average they take more time in moving to and away from their seat compared to other passengers.

Special passengers have different categories, but in this study, they are dealt as a single group and only their total amount for arrival and departure is given for the machine learning algorithm. When a special passenger is on board a plane, he/she may need a wheelchair to board or deboard the airplane and it might create an increase in duration of the boarding or deboarding processes of a turnaround. Figure 4 indicates that a situation where more than one special passenger is on board the aircraft is rare.

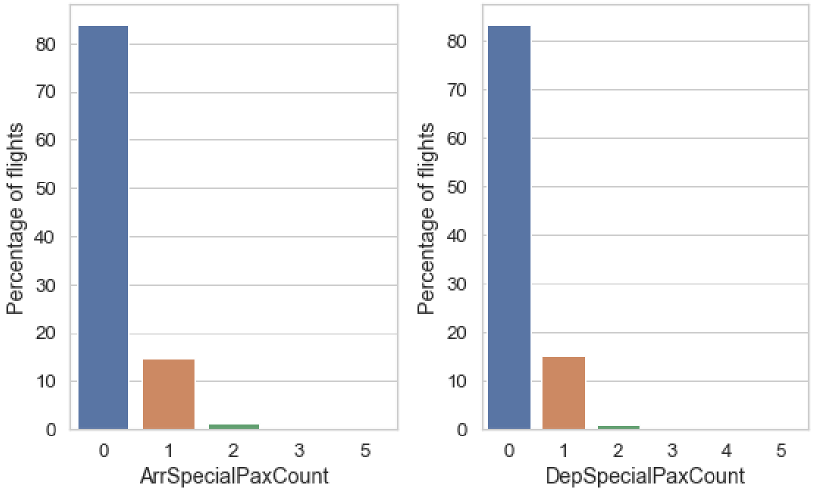


Figure 4: Arriving and departing number of special passengers

Month, *Season* and *TimeOfDay* refer to the moment in which the turnaround will take place. Month is a numerical value of a month and therefore quite self-explanatory. Instead, Season and TimeOfDay refer to variables where months and hours of day are grouped. Season consists of three groups which are winter season, midseason and summer season. Months are divided into seasons so that months from November to February are considered as winter, months from May to August are considered as summer and the remaining months in between constitute the midseason.

Time of day consists of three groups which are morning, midday and evening. Hours from 4 to 9 UTC are considered as morning, hours from 10 to 14 UTC are considered as midday and hours from 15 to 19 UTC are considered as evening. Remaining hours form the night group but there are no turnarounds in airport A during that time. The reason for choosing variables Season, TimeOfDay and Month is the objective to capture possible patterns in airport operations, which are not evident or for which data is not available, but which are dependent on these variables and have an effect on the turnaround time. Such changes in airport operations could be the number of ground or handling personnel working at a certain moment or the effect of temperature on the ground handling procedures which changes according to time of day and month of the year.

As indicated by Table 5 previously, Season variable works only as a supplementary variable in the analysis and the Month feature, on the other hand, is expected to capture patterns that are related to different times of the year and is eventually given as an input for XGBoost algorithm. In the Figure 5 below the division of flights between different times of day and seasons of the year are displayed. Based on the figure, the division of flights between aircraft variants is more or less the same with the exception of morning flights and flights during winter season when A320 variant has most flights.

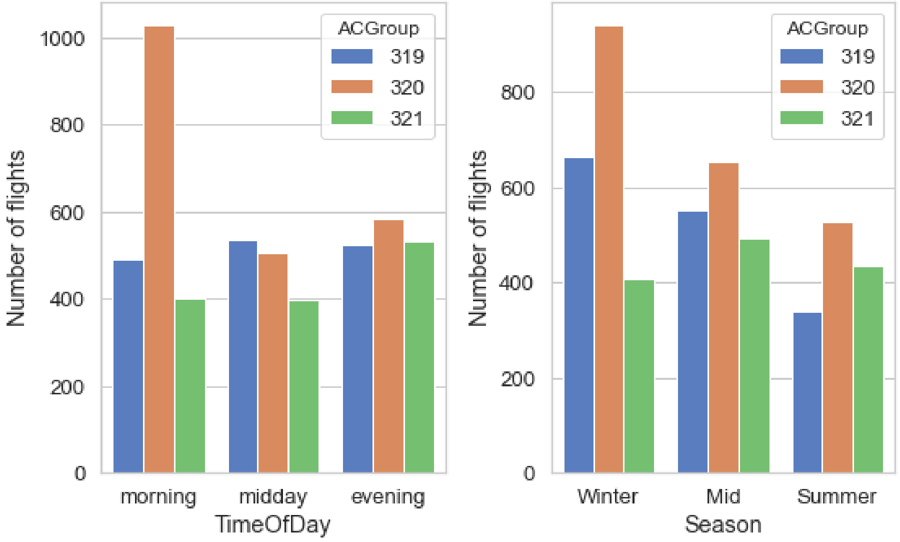


Figure 5: Distribution of flights for different times of day and different seasons for different aircraft variants

Features *BagsIn* & *BagsOut, CargoIn* & *CargoOut* and *MailIn* & *MailOut* refer to the inbound and outbound flow of material which is measured in kilograms and is loaded into the cargo compartments. Bags are related to number of passengers as passengers bring bags with them to the airport whereas cargo is flow of material which is not directly related to passengers. Below in Figure 6 is shown the distribution of bags in kilograms for arriving and departing flights for different aircraft variants. Based on the figure, it can be noted that distributions for departing bags are similar between different variants whereas for arriving bags A321 has a more dispersed distribution compared to A320 and A319.

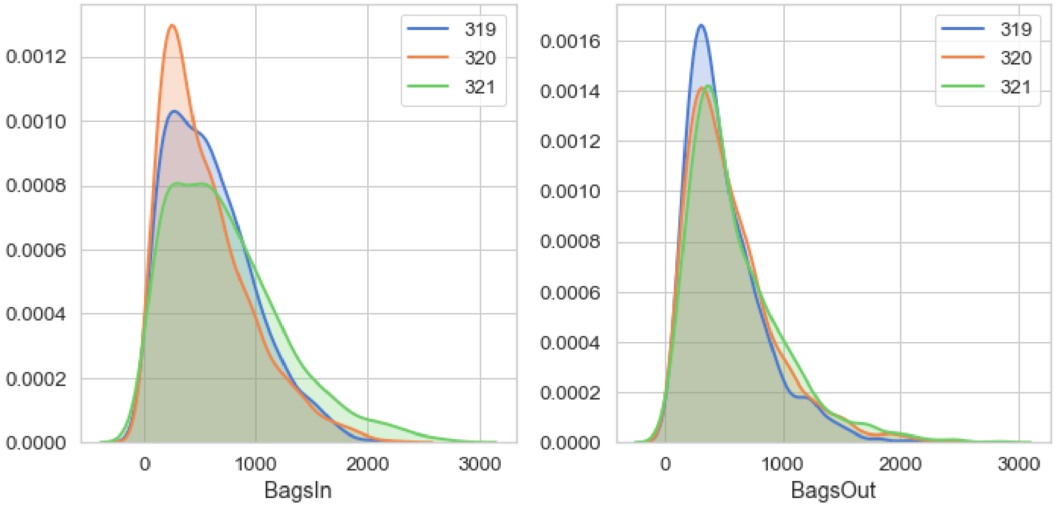


Figure 6: Distribution of bags in kilograms for arrival and departure flights for different aircraft variants

Figure 7 displays information regarding the percentage of flights with inbound or outbound cargo for different aircraft variants. Based on the figure, inbound cargo is most probably carried by A320 whereas less likely variant to be carrying outbound cargo is A321. However, the differences between variants for carrying cargo are not large.

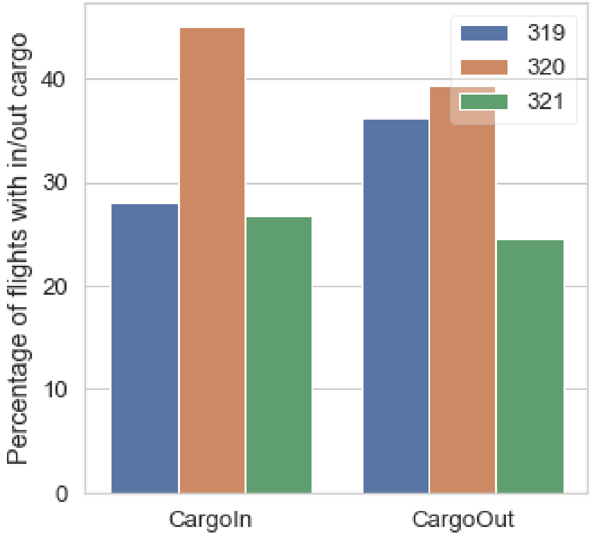


Figure 7: Percentage of flights with inbound or outbound cargo for different aicraft variants

Based on the dataset, mail has been carried to airport A on 70 flights with a maximum amount of 48 kilograms and from airport A only on 7 flights with a maximum amount of 20 kilograms. In the case of airport A, aircraft carrying mail is a rare event and the amount of carried mail is extremely small compared to other flow of people and material. However, mail variables will be included in the dataset as their effect on turnaround durations is unknown.

FuelNeed refers to the need of fuel in kilograms. If fuelling need has a value of zero, it means fuelling has not been performed. Out of 5010 rows in the preprocessed dataset, fuelling has been performed only on 19 turnarounds and for those turnarounds the average fuelling is 1604 kilograms. Out of 19 turnarounds, 5 have been for A319, 8 for A320 and 6 for A321. Therefore, also fuelling is a rare event for turnarounds in airport A and there is no significant difference between the number of fuellings performed for different aircraft variants.

Deice feature indicates whether a deicing has been performed. Dataset reveals the need for deicing in the past but when new predictions are made, it might be hard to know in advance whether deicing will be performed or not. Environmental variables such as temperature and amount of precipitation could be used in the anticipation of whether deicing is needed but that is not certain and is not in the scope of this thesis. Deicing can be performed in two steps where the first step consists of frost and ice removal and the second step, also known as anti-icing, prevents new frost and ice from accumulating on the critical surfaces of the aircraft (Norin et al., 2012).

In this study deicing steps are not separated but instead deicing is treated as one process for two reasons. Firstly, from predictive perspective it is difficult to know in advance whether deicing will be performed at all in the coming turnaround and even harder to know whether one or two steps are required in a single turnaround. Secondly, from a descriptive perspective it is beneficial to know the total effect of deicing on turnarounds. Number of steps in deicing affect the length of the deicing but assuming both one and two step deicings are included in the dataset, it is possible to identify the average total effect and there is no need to make distinction between different deicings. As deicing is considered in this thesis as a Boolean value with two options, prediction for turnaround times in the future can be made both with and without deicing.

Scheduled TAT and *Available TAT* are concepts which have been discussed in section 2.2. Scheduled turnaround time is the time span from the scheduled arrival time at the gate to the scheduled departure time when the aircraft is expected to be ready for pushback. Available turnaround time is the time span from the actual arrival time on the gate or stand to the scheduled departure time. Figure 8 below indicates the distributions for Scheduled TAT for different aircraft variants and times of day. Based on the figure it can be pointed out that most typical scheduled turnaround time for different aircraft variants and times of day is 30 minutes.

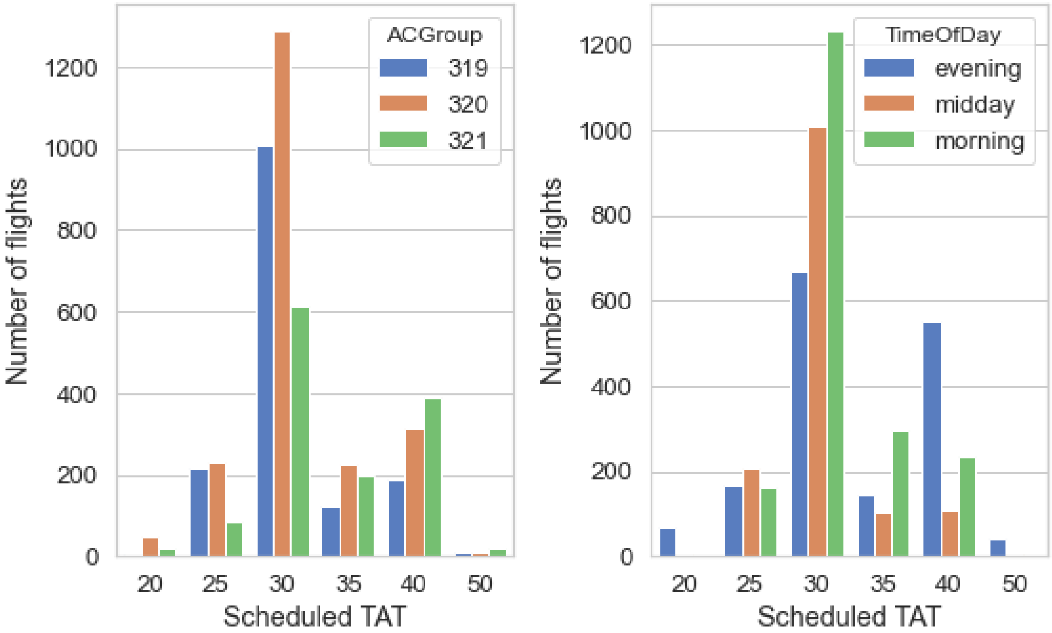


Figure 8: Distributions of scheduled turnaround times in minutes for different aircraft variants and different times of day

Figure 9 below indicates the distributions of Available TAT values in minutes for different aircraft variants. Based on the figure, distributions seem to be really similar between aircraft variants. A long tail on the left side of the curve indicates that there are turnarounds with a large delay with a large negative value for available turnaround time.

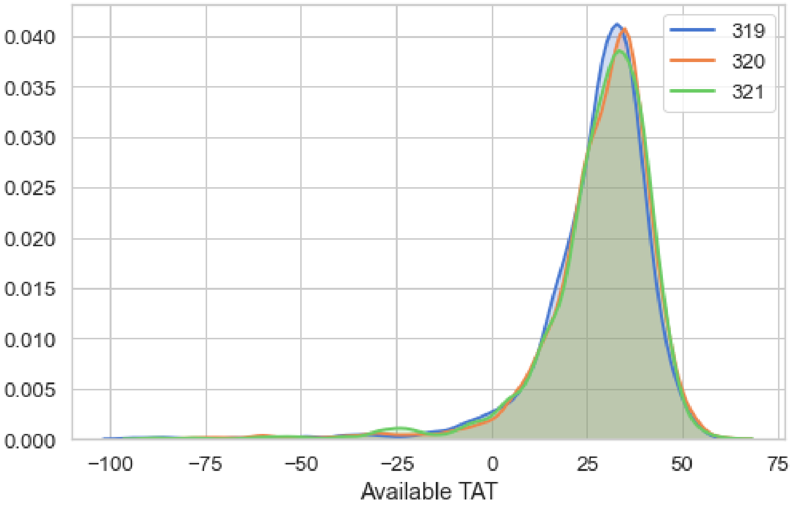


Figure 9: Distribution of available turnaround times in minutes

TurnaroundTime is the target variable and the distribution of turnaround times is indicated in Figure 10. Based on Table 6 median for A319 is 32 minutes whereas for A320 it is 33 minutes and A321 34 minutes. Aircraft variants A320 and A321 have higher densities between interval 40 and 50 minutes according to Figure 10. This could be due to the reason that on average they carry more passengers which in turn results in longer boarding times.

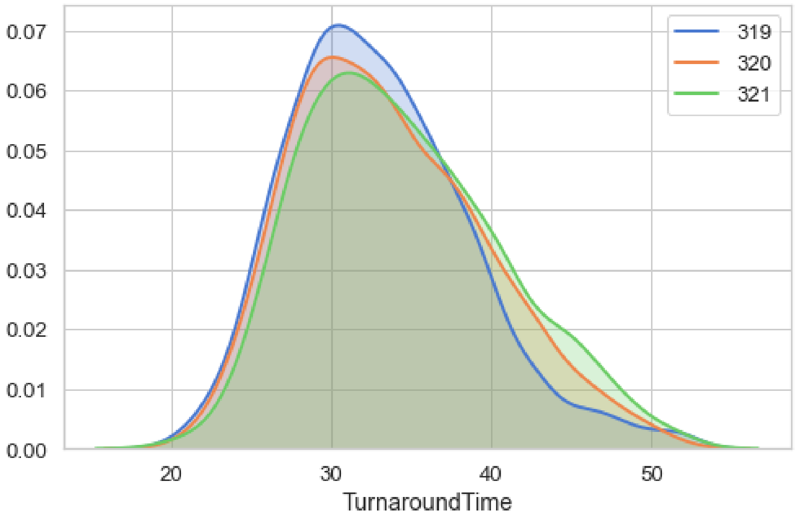
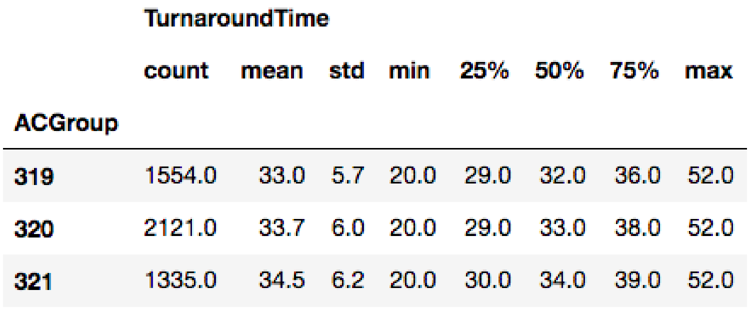


Figure 10: Distribution of turnaround times in minutes

Table 6: Statistics of turnarounds for different aircraft variants



	TurnaroundTime							
	count	mean	std	min	25%	50%	75%	max
ACGroup								
319	1554.0	33.0	5.7	20.0	29.0	32.0	36.0	52.0
320	2121.0	33.7	6.0	20.0	29.0	33.0	38.0	52.0
321	1335.0	34.5	6.2	20.0	30.0	34.0	39.0	52.0

TAT difference feature is the difference between features TurnaroundTime and Scheduled TAT. It is included as a supplementary variable as it indicates whether actual turnaround time has been larger than scheduled turnaround time. It is worth to note that TAT difference does not indicate whether a flight has been late as a flight can be significantly late and still have similar durations for turnaround time and scheduled turnaround time. Figure 11 below indicates that for TAT difference the dispersion of the distribution is narrower for A319 compared to A320 and A321, of which A321 has the most dispersed distribution. A321 also has the largest density between interval -15 and -5 minutes. This means that A321 has had more often turnaround durations where the actual turnaround time has been from -15 minutes to -5 minutes shorter compared to the scheduled turnaround time. Table 7 indicates that for A319 turnaround durations have been closer to scheduled turnaround compared to two other aircraft variants.

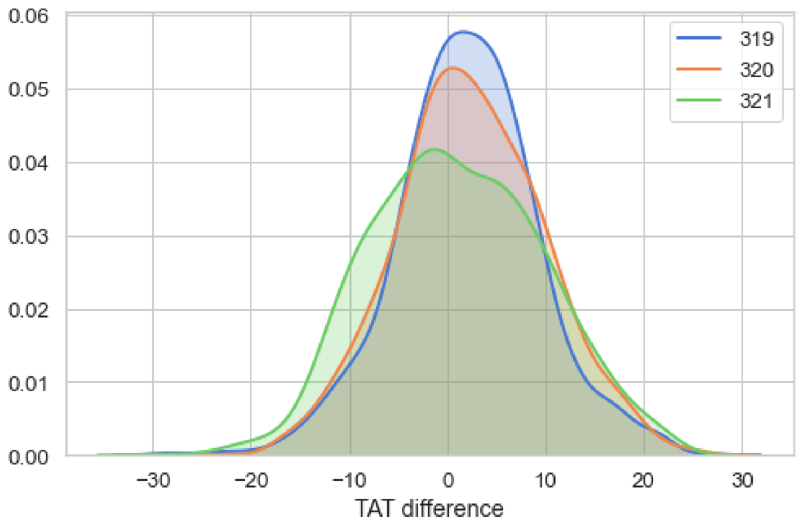
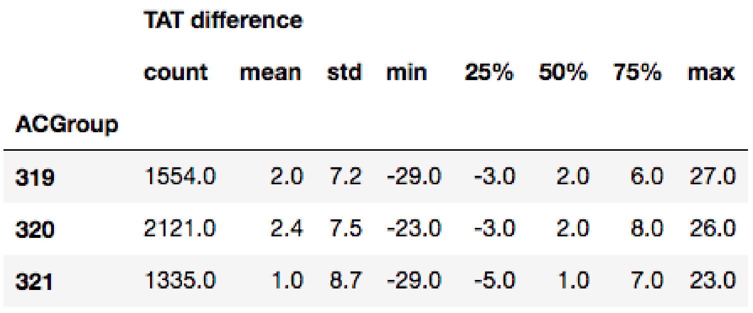


Figure 11: Distributions of differences between actual and scheduled turnaround times in minutes

Table 7: Statistics of feature TAT difference for different aircraft variants



	TAT difference							
	count	mean	std	min	25%	50%	75%	max
ACGroup								
319	1554.0	2.0	7.2	-29.0	-3.0	2.0	6.0	27.0
320	2121.0	2.4	7.5	-23.0	-3.0	2.0	8.0	26.0
321	1335.0	1.0	8.7	-29.0	-5.0	1.0	7.0	23.0

Data regarding the situation where simultaneous boarding and fueling is performed would be useful as simultaneous operation could diminish the effect of fueling on turnaround time in airport A. However, in this study such data is not available, and it is not possible to make assumptions about how fuelling is performed in airport A for this dataset. Next, the preprocessing performed for the dataset will be discussed.

4.7 Data preprocessing

In this section data preprocessing steps, which have been performed to prepare the data for the machine learning algorithm XGBoost, will be discussed. Data preprocessing steps include the removal of rows containing missing values, removal of outliers, removal of flights with specific delay codes and finally the encoding for categorical data. As data has been collected from multiple databases, different database tables have been joined together to create the final dataset for this study.

Even though XGBoost has the support for sparsity-aware split finding (Chen & Guestrin, 2016), which means that it accepts rows with missing values, a preprocessing step has been included where rows with missing values are removed before training the machine learning model with the dataset.

There can be many reasons for the existence of outliers in data and in general they can be considered as samples that are far from the mainstream of the data (Iglewicz & Hoaglin, 1993). Proper recognition and handling of outliers are important preprocessing steps as outliers in data might cause reduction in the R^2 (James et al., 2013). Boxplot method, originally presented by Tukey (1977), is a simple and robust way to detect univariate outliers from data. In this analysis, boxplot method is used in the detection and removal of outliers from the turnaround time variable which is the target variable in the data. Another measure to recognize outliers is the assessment based on domain knowledge which is performed on extreme values of other features of the dataset. In addition to boxplots, histograms and other means of data visualization can be helpful in determination of outliers and unusual values in the data. According to Kuhn and Johnson (2013), the definition of an outlier may be difficult even with good understanding of the phenomenon behind the dataset, but nevertheless, it is often possible to identify unusual values by inspecting individual extreme figures.

Dawson (2011) describes that boxplot is a visualization tool which shows the median together with 3rd and 1st quartiles of the data. In addition, the so-called fences, or boundaries, are set to enable flagging of outliers and they are usually determined by multiplication of the difference between 3rd and 1st quartiles with a value of 1.5. Based on these boundaries, he notes, boxplots indicate the minimum and maximum values of the data that are inside the fences with whisker lines. Single points which are not part of the box or the whiskers are considered as outliers (Dawson, 2011). Figure 12 below visualizes the boxplot model and Figure 13 displays boxplots for different continuous variables after the preprocessing steps have been performed excluding the features PaxOnBoardIn, PaxOnBoardOut, CargoIn and CargoOut.

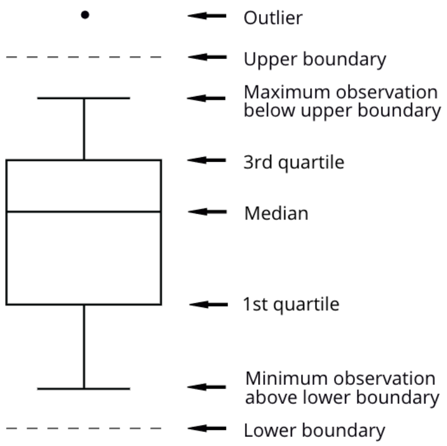

Outlier
Upper boundary
Maximum observation below upper boundary
3rd quartile
Median
1st quartile
Minimum observation above lower boundary
Lower boundary

Figure 12: Elements of a boxplot (Kwak & Kim, 2017)

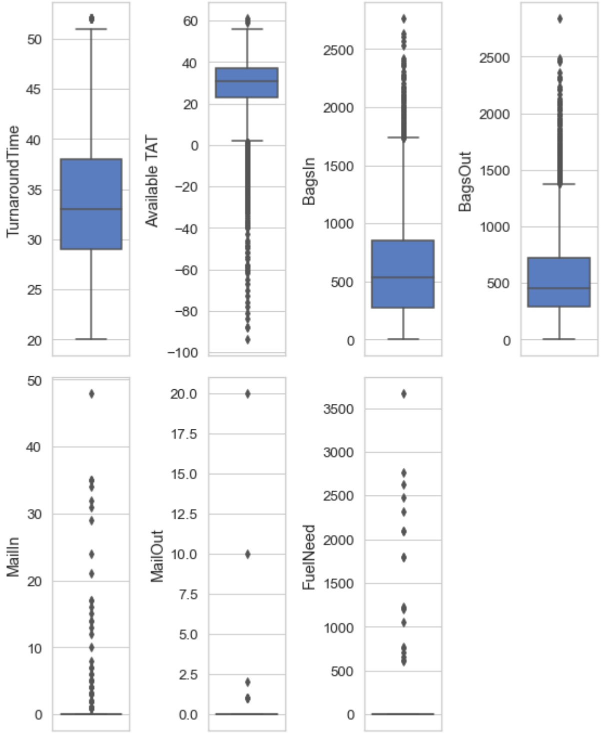


Figure 13: Boxplots for preprocessed continuous variables

Boxplot method described above was performed for the TurnaroundTime variable and Figure 13 indicates that only a single observation is left outside the whiskers of that boxplot. For other variables domain knowledge was used to remove most extreme observations which could be considered as outliers. For certain variables such as MailIn, MailOut and FuelNeed it is difficult to determine a threshold for outliers. This is due to the fact that only a small number of rows exist in the dataset where values for said variables were other than zero and there is no prior knowledge whether this small group of rows could prove to be significant in the prediction of a turnaround process.

Another way of recognizing unwanted samples in our training data is to inspect if there exist specific delay codes for turnarounds which are late from their original schedule. Delay code specifies the reason for the occurred delay and the delay code system is standardized by IATA (Eurocontrol, 2020). Goal of the inspection of delay codes is to filter out samples with delays which could not be predicted by machine learning algorithm with the given dataset. Some examples of such delay causes are technical issues, sudden illnesses of personnel or restrictions placed by the air traffic controller. List of delay codes which are used as a filter to remove samples can be found from Appendix B. However, some delay causes such as aircraft deicing will be included in the final dataset as removal of such observations could distort the effect of deicing on turnaround durations in general.

Removal of samples with specified delay causes, concerns only turnarounds which have been late from their scheduled departure time. Detection of outliers with the boxplot method enables the removal of samples which may not have been late but might have otherwise exceptional value in terms of turnaround time or values of the feature variables. Together these two preprocessing steps make the data more generalizable as the remaining dataset consists of rows with turnaround times that the independent variables are better capable of explaining.

For categorical variables, such as ACGroup and Season, one-hot encoding was performed. In one-hot encoding categorical variables will be transformed into dummy variables where each category has its own column (Kuhn & Johson, 2013). In columns which have been encoded by using one-hot encoding, each cell has either a value one or zero, depending on whether that variable is present in that specific row. Next sections will describe the process of feature selection and identification of interaction and collinearity between variables as well as the concept of cross-validation for hyperparameter tuning and metrics for assessment of the final model.

4.8 Feature selection

Feature selection is the process of determining which subset of possible features for the final machine learning model would be the most potential one. According to Dash & Liu (1997), in real-world applications the relevancy of features in their contribution to the prediction of the target variable is many times unknown a priori. Domain knowledge can help in the preliminary determination of features that are required for the predictive capability, but different feature selection methods bring further evidence about the true contribution of those features. Guyon & Elisseef (2003) note, that inclusion of only relevant features provides faster and more cost-efficient predictive models and relevant features also can provide better understanding of the phenomenon that is being inspected.

In this study two different feature selection methods are used to determine the best subsets, and their results are compared with each other. These methods include the wrapper method Recursive Feature Elimination (RFE) (Granitto et al., 2006) and the feature importance method provided by SHAP. Granitto et al. (2006) explain that RFE starts with fitting the machine learning model for data with all the features. After fitting the model, feature importance values are calculated, and the less relevant feature is removed followed by a re-fitting of the model with new subset of features. This process is recursively followed until the desired number of features is remaining, according to Granitto et al. (2006).

As there is no certainty about the optimal number of features, as part of the feature selection process, all the different feature subset sizes from 1 to the total number of features will be tested with the RFE algorithm. By using cross-validation, accuracy score will be calculated for all the models given by the RFE and the choice for the final subset size is made based on those accuracies.

In addition to RFE, the concept of Pareto principle (Pareto, 1896) will be applied to the feature subset selection that takes advantage of the feature importance values calculated by SHAP. In this case, the Pareto principle is applied so that by starting from the features with the largest feature importance values, features will be added to a subset of features until their cumulative feature importance accounts for 80% of the total cumulative feature importance. As a benchmark, this subset of features is compared against the subset which was acquired by using RFE.

According to Guyon & Elisseef (2003) the RFE method is computationally intensive due to its exhaustive nature. However, in this study the number of features can be considered more or less moderate and therefore the computational point of view is not relevant. The machine learning model at the core of the feature selection method RFE does not need to be the same as the final model used for the turnaround predictions (Granitto et al., 2006).

In addition to the full set of input features, three different subsets will be initially provided for the algorithm. Two subsets will be selected by using the RFE method. One is provided by the RFE using a random forest algorithm and another one is provided by using XGBoost with default hyperparameter values. Third subset is identical to the full set of features excluding features Available TAT and Scheduled TAT. Argument in favor of the third subset is that by excluding information related to turnaround schedules, XGBoost algorithm needs to derive the turnaround time solely based on other features such as passenger count and amount of cargo which change on daily basis and are not affected by historical turnaround times. This way XGBoost is not able to put too much emphasis on scheduled times which, to a certain extent, correlate with the actual turnaround times. Eventually SHAP will be used to provide feature importance values by using the fully trained and tuned XGBoost model, which can also be used as a benchmark in comparison to the subsets originally selected.

4.9 Interaction and collinearity

As there are multiple variables used for the prediction of the turnaround duration, some variables might demonstrate interaction which means that the effect of one independent variable to the target variable may significantly change at the presence of another independent variable (Cortina, 1993). One case presented by Guyon and Elisseef (2003) is when a single variable might be useless based on its predictive contribution alone but together with some other variables it might add a significant improvement into the predictive performance. Collinearity on the other hand is the correlation between independent variables (Cortina, 1993).

Chen, He, Benesty and Tang (2018) point out that decision trees and boosted trees are robust to correlated features, that is, collinearity. Also, due to the recursive nature of the RFE method, features that demonstrate interaction effects will be included in the subsets of features if the effects are strong enough and they are conveyed through changes in feature importance values (Granitto et al., 2006). However, it is hard to determine when the interaction effect is strong enough between features that they will surely be included in the final subset selected by the feature selection methods. Based on this discussion, it can be stated that information regarding potential connections between features, and their interactions, is valuable in itself to improve the interpretability and understanding of a turnaround process.

Tools that are used to reveal global interaction effects in this study are the calculation of SHAP interaction values (Lundberg et al., 2019), which were discussed already in the earlier section, and the calculation of the H-statistic (Friedman & Popescu, 2008). One method to uncover multicollinearity between variables is to calculate the variable inflation factors (VIF) (Mansfield & Helms, 1982).

H-statistic, which was originally introduced by Friedman and Popescu (2008), can be used to measure interactions. In this study especially the property of calculating two-way interactions between a pair of features will be used. Molnar (2019) describes the mathematical formula for calculating H-statistic which is presented below as Equation 4. The main idea of H-statistic is that it calculates the difference between partial dependence of two features together ($PD_{jk}(x_j, x_k)$) and partial dependence of two features individually ($PD_j(x_j)$ and $PD_k(x_k)$). If interaction between inspected features does not exist, H-statistic has a value of 0 and if the prediction is only affected by the interaction, the value of the H-statistic is 1 (Molnar, 2019). Formula presented by Equation 4 is calculated for all data points. At the moment of writing this thesis, H-statistic Python library provides support only for gradient boosting regressor which will be used to calculate partial dependencies (Haygood, 2017). Calculation of H-statistic is presented as

$$H_{jk}^2 = \sum_{i=1}^{n}[PD_{jk}\left(x_j^{(i)}, x_k^{(i)}\right) - PD_j(x_j^{(i)}) - PD_k(x_k^{(i)})]^2 / \sum_{i=1}^{n} PD_{jk}^2(x_j^{(i)}, x_k^{(i)}). \quad (4)$$

Daoud (2017) presents that variance inflation factors (VIF) indicates whether correlation between independent variables exist. He suggests that VIF can be calculated, by linearly regressing i-th variable against all other variables, presented as

$$VIF = \frac{1}{1-R_i^2} \, . \qquad (5)$$

VIF value of 1 indicates no correlation, values between 1< and ≤5 indicate that moderate correlation exists and VIF value over 5 indicates a high correlation for that specific feature compared to other features, based on Daoud (2017).

4.10 Cross-validation and hyperparameter tuning

Cross-validation is a method for the estimation of prediction error where separate validation set is used to assess how a machine learning model will generalize to unseen data, according to Friedman et al. (2001). They note that k-fold cross-validation, on the other hand, is cross-validation where data is divided into certain amount of folds. Model is fitted with k-1 parts of the data and the remaining part is used to calculate the prediction error. This process is followed by training another model with a new set of k-1 parts of the data, and again, the remaining part is used to calculate error. The fitting of models continues until all the folds have been used as a validation set and finally the prediction error of different models is combined to demonstrate the generalization error of a trained model for new data, presented by Friedman et al. (2001).

The justification for the use of k-fold cross-validation in this thesis is that it is useful in situations when data is scarce and not enough data exists to properly assess the prediction accuracy with a distinct validation set during training. However, a separate test set of unseen data is provided for the fully trained model and is analyzed with interpretability methods.

Based on Kuhn and Johnson (2013), a general approach for the search of best hyperparameters is to create a set of candidate values, generate metrics for the performance of the model with different hyperparameter combinations and select the best performing set. In this thesis, due to the multitude of hyperparameters and their different combinations, nested cross-validation will be used to select best performing hyperparameters for different feature sets.

Nested cross-validation consists of inner and outer cross-validation loops where outer loop is divided into training and testing parts and the training part is furthermore divided into training and validation sets for the inner loop (Wainer & Cawley, 2018). In this case inner loop is used for the tuning of the hyperparameters of the XGBoost model and outer loop is used to estimate the error of the selected hyperparameters provided by the inner loop. In practice this means that inner loop trains models with different hyperparameter options, chooses best parameters based on the validation set of the inner loop and returns the model with best hyperparameters for the outer loop for evaluation of the generalization.

4.11 Metrics for model assessment

One way to measure the performance of a regression model is the mean error between the predictions and actual target values. According to Willmott and Matsuura (2005), mean absolute error (MAE) is a natural and unambiguous way to measure average error in a training and test set and therefore MAE will be used to assess the final model in this thesis. Formula for calculating MAE, provided by Willmott and Matsuura (2005), is presented as

$$MAE = n^{-1} \sum_{i=1}^{n} |P_i - O_i|. \quad (6)$$

In Equation 6, P_i corresponds to the value of the i-th prediction whereas O_i corresponds to the value of the i-th observation, corresponding to the actual duration of the i-th turnaround. It is worth to note that during training, XGBoost uses mean squared error in the objective function as discussed in section 4.3.2. However, cross-validation uses MAE in the estimation of the generalization performance in this study.

From operative perspective, performance of the model assessed through the mean error value illustrates the general capability of the model but in addition it is beneficial to have information regarding the goodness-of-fit. Based on Nagelkerke (1991), goodness-of-fit can be measured with R^2 ,which is also known as the explained variance. R^2 measures the proportion of the variation of the dependent variable that is explained by the independent variables and it is a dimensionless value. In this case R^2 conveys information on how much of the variation of turnaround times is explained by the input features such as passenger count. Formula for calculating explained variance in the Scikit-learn library (Scikit-learn, 2020) is presented as

$$Explained\ variance\ (R^2) = 1 - \frac{Var\{y-\hat{y}\}}{Var\{y\}}. \quad (7)$$

Value of the explained variance close to 1 implies that predictions derived from independent variables are able to perfectly explain variations in the actual values. Lower values, on the other hand, imply an imperfect capability in explaining the variations.

Explained variance is a useful measure from operative perspective as it reveals how well the gathered data is able to explain the inspected phenomenon. In an optimal situation the gathered dataset contains all the variables that are suspected to be able to determine the variations in the target variable. If the explained variance is close to 1, it is possible to state that the correct features have been selected and these give a reasonable understanding of the phenomenon. Instead, if the explained variance is low, it can be concluded that some aspects of the phenomenon are not captured by the data which would be needed to make even more accurate predictions.

The metrics discussed in this section for model performance assessment can be calculated for XGBoost models trained with different feature subsets. By comparing these performance figures for different subsets, conclusions can be drawn on the predictive capabilities of individual features. Combined with information provided by SHAP, conclusions can be further refined and used in future research as well as in operational decision making for potential improvements related to the turnaround process in airport A. In the next chapter, results for the analysis on turnaround process in airport A will be reviewed.

5 Results

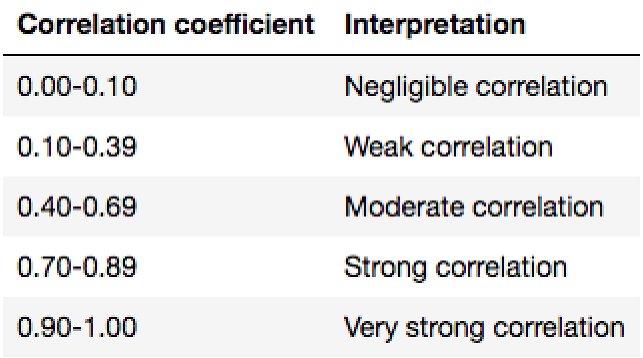
This chapter contains a walkthrough on the results of the analysis and predictive modeling of the turnaround process. First, the correlations between variables and inspection for multicollinearity will be reviewed. Second, bi-variate visualizations for features, which are mainly selected based on the correlation results, will be provided. That is followed by the review of interaction results provided by the calculation of the H-statistic. After preliminary interaction determination, results related to XGBoost performance assessment are reviewed. Finally, predictions provided by the trained machine learning model will be used in the calculation of SHAP values. SHAP values are used to visualize both the global and local effect of features in the turnaround duration determination. In addition, SHAP interaction values are calculated and used to select visualizations for feature pairs. At the end of this chapter, a summary will tie together the most essential findings of this analysis.

5.1 Correlations

Correlation is a measure of association between variables (Schober, Boer & Schwarte, 2018). In this section, correlation is used as a tool to reveal connections between variables and these connections will be furthermore used to determine necessary visualizations. Schober et al. (2018) point out that Pearson correlation, which is used in this study, is able to reveal linear relationships between variables and that the correlation coefficient indicates both strength and direction of the relationship. Value of 1 or -1 indicates a perfect correlation and a value close to 0 implies that no correlation exists.

Table 8 provides thresholds for different correlation coefficients that can be used in the interpretation of the strength of the correlation. It is worth to note, however, that correlation coefficient close to 0 does not prove that two variables are not connected in any way. Weak or negligible correlation shows that two inspected variables may not have a linear relationship but might otherwise be related. Schober et al. (2018) discuss that absolute limits for separating weak correlation from moderate or moderate from strong is hard to determine and instead the strength of correlation through coefficients should be considered in the context of the scientific question. In addition, they (Schober et al., 2018) undermine that correlation does not indicate a causal relationship between two variables.

Table 8: Interpretation of the correlation coefficient (Schober, Boer & Schwarte, 2018)

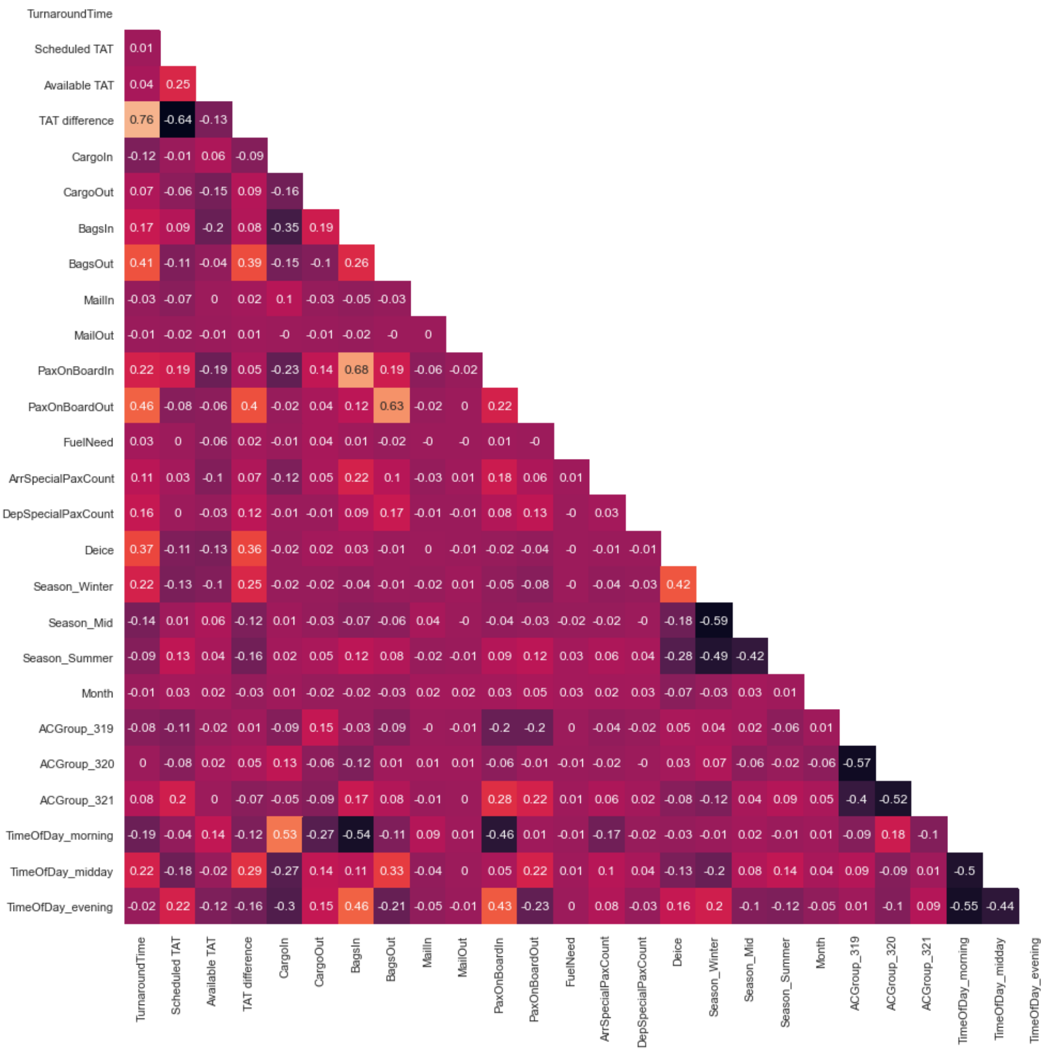


Correlation coefficient	Interpretation
0.00-0.10	Negligible correlation
0.10-0.39	Weak correlation
0.40-0.69	Moderate correlation
0.70-0.89	Strong correlation
0.90-1.00	Very strong correlation

The analysis of correlation coefficients will proceed as follows. First, correlation coefficients are calculated by using the correlation function from the Pandas library (Pandas, 2020) and they are shown in Table 9 below. Heuristically determined thresholds of 0.2 and -0.2, which according to Table 8 present approximately a middle value for weak correlations, are used to filter out weaker correlations. This means that correlation coefficients between values -0.2 and 0.2 are not included in the further analysis. Such thresholds were determined partly as there is no further scientific evidence on this topic to specify more accurate values.

Second, irrelevant correlations such as correlation between features Month and Season or TAT difference and TurnaroundTime are removed as comparison of correlation between such features is not either meaningful or either one of the features is derived from the other. Finally, remaining coefficients are processed and their usability for gaining understanding of the turnaround process is assessed.

Table 9: Correlations between variables



	TurnaroundTime	Scheduled TAT	Available TAT	TAT difference	CargoIn	CargoOut	BagsIn	BagsOut	MailIn	MailOut	PaxOnBoardIn	PaxOnBoardOut	FuelNeed	ArrSpecialPaxCount	DepSpecialPaxCount	Deice	Season_Winter	Season_Mid	Season_Summer	Month	ACGroup_319	ACGroup_320	ACGroup_321	TimeOfDay_morning	TimeOfDay_midday	TimeOfDay_evening
TurnaroundTime																										
Scheduled TAT	0.01																									
Available TAT	0.04	0.25																								
TAT difference	0.76	-0.64	-0.13																							
CargoIn	-0.12	-0.01	0.06	-0.09																						
CargoOut	0.07	-0.06	-0.15	0.09	-0.16																					
BagsIn	0.17	0.09	-0.2	0.08	-0.35	0.19																				
BagsOut	0.41	-0.11	-0.04	0.39	-0.15	-0.1	0.26																			
MailIn	-0.03	-0.07	0	0.02	0.1	-0.03	-0.05	-0.03																		
MailOut	-0.01	-0.02	-0.01	0.01	-0	-0.01	-0.02	-0	0																	
PaxOnBoardIn	0.22	0.19	-0.19	0.05	-0.23	0.14	0.68	0.19	-0.06	-0.02																
PaxOnBoardOut	0.46	-0.08	-0.06	0.4	-0.02	0.04	0.12	0.63	-0.02	0	0.22															
FuelNeed	0.03	0	-0.06	0.02	-0.01	0.04	0.01	-0.02	-0	-0	0.01	-0														
ArrSpecialPaxCount	0.11	0.03	-0.1	0.07	-0.12	0.05	0.22	0.1	-0.03	0.01	0.18	0.06	0.01													
DepSpecialPaxCount	0.16	0	-0.03	0.12	-0.01	-0.01	0.09	0.17	-0.01	-0.01	0.08	0.13	-0	0.03												
Deice	0.37	-0.11	-0.13	0.36	-0.02	0.02	0.03	-0.01	0	-0.01	-0.02	-0.04	-0	-0.01	-0.01											
Season_Winter	0.22	-0.13	-0.1	0.25	-0.02	-0.02	-0.04	-0.01	-0.02	0.01	-0.05	-0.08	-0	-0.04	-0.03	0.42										
Season_Mid	-0.14	0.01	0.06	-0.12	0.01	-0.03	-0.07	-0.06	0.04	-0	-0.04	-0.03	-0.02	-0.02	-0	-0.18	-0.59									
Season_Summer	-0.09	0.13	0.04	-0.16	0.02	0.05	0.12	0.08	-0.02	-0.01	0.09	0.12	0.03	0.06	0.04	-0.28	-0.49	-0.42								
Month	-0.01	0.03	0.02	-0.03	0.01	-0.02	-0.02	-0.03	0.02	0.02	0.03	0.05	0.03	0.02	0.03	-0.07	-0.03	0.03	0.01							
ACGroup_319	-0.08	-0.11	-0.02	0.01	-0.09	0.15	-0.03	-0.09	-0	-0.01	-0.2	-0.2	0	-0.04	-0.02	0.05	0.04	0.02	-0.06	0.01						
ACGroup_320	0	-0.08	0.02	0.05	0.13	-0.06	-0.12	0.01	0.01	0.01	-0.06	-0.01	-0.01	-0.02	-0	0.03	0.07	-0.06	-0.02	-0.06	-0.57					
ACGroup_321	0.08	0.2	0	-0.07	-0.05	-0.09	0.17	0.08	-0.01	0	0.28	0.22	0.01	0.06	0.02	-0.08	-0.12	0.04	0.09	0.05	-0.4	-0.52				
TimeOfDay_morning	-0.19	-0.04	0.14	-0.12	0.53	-0.27	-0.54	-0.11	0.09	0.01	-0.46	0.01	-0.01	-0.17	-0.02	-0.03	-0.01	0.02	-0.01	0.01	-0.09	0.18	-0.1			
TimeOfDay_midday	0.22	-0.18	-0.02	0.29	-0.27	0.14	0.11	0.33	-0.04	0	0.05	0.22	0.01	0.1	0.04	-0.13	-0.2	0.08	0.14	0.04	0.09	-0.09	0.01	-0.5		
TimeOfDay_evening	-0.02	0.22	-0.12	-0.16	-0.3	0.15	0.46	-0.21	-0.05	-0.01	0.43	-0.23	0	0.08	-0.03	0.16	0.2	-0.1	-0.12	-0.05	0.01	-0.1	0.09	-0.55	-0.44	

The most important question related to Table 9 is whether there is correlation between TurnaroundTime and other variables. Feature PaxOnBoardOut has a moderate correlation with TurnaroundTime (0,46) as well as the feature BagsOut (0,41). Feature PaxOnBoardIn constitutes only a weak correlation with TurnaroundTime (0,22). Table 9 indicates a correlation coefficient of 0,68 between features PaxOnBoardIn and BagsIn which is reasonable as presumably more passengers bring more bags with them. Similar correlation applies for features PaxOnBoardOut and BagsOut.

TurnaroundTime has a correlation coefficient of 0,37 with feature Deice which is logical as deice in general is not a default process in a turnaround. Deice correlates positively with Season_winter (0,42) and negatively with Season_summer (-0,28), whereas TurnaroundTime and Season_winter have a correlation coefficient of 0,22. The positive correlation between

TurnaroundTime and Season_winter could be partly due to deicing which occurs mainly during winter, as the correlation between different seasons and deicing suggests, but it cannot be deduced based on the information provided by Table 9.

There exists a weak positive correlation between TurnaroundTime and the feature TimeOfDay_midday (0,22). This suggests that flights operated during midday hours (10-14 UTC) have a longer turnaround duration. Reason for this, however, is not evident.

TAT difference is a feature that is derived by subtracting Scheduled TAT from TurnaroundTime. It shows similar correlation coefficients with features PaxOnBoardOut (0,4), BagsOut (0,39), Deice (0,36), Season_winter (0,25) and TimeOfDay_midday (0,29) as the feature TurnaroundTime. This observation emphasizes the pattern that not only the presence of these features increases the duration of a turnaround but in addition the turnarounds are longer than planned and there is a higher chance for a late departure.

Observations from the Table 9 correlations related to aircraft variants are that PaxOnBoardIn and PaxOnBoardOut correlate positively with the feature ACGroup_321 (coefficients 0,28 and 0,22) whereas PaxOnBoardIn and PaxOnBoardOut have a negative correlation with ACGroup_319 (coefficients -0,2 and -0,2). This might be due to the fact that different aircraft variants have different passenger capacities and presumably larger capacities are used only when enough demand exists. However, the weak correlation effect between inbound and outbound passengers and different aircraft variants is not conveyed into significant correlation between ACGroup features and turnaround time.

Weak correlations between features BagsIn and BagsOut (0,26), as well as between features PaxOnBoardIn and PaxOnBoardOut (0,22), demonstrate an example where an actual dependency between these features is difficult to justify. It is hard to establish a connection between the amount of arriving passengers and departing passengers. However, their correlation could be partly explained with a third variable ACGroup, as discussed above, where greater passenger and cargo capacity would result in more passengers and bags being transported to and from airport A.

TimeOfDay features correlate with the flow of people, bags and cargo. A certain pattern of those correlations can be seen as TimeOfDay_morning has a positive correlation with CargoIn (0,53) but a negative correlation with CargoOut (-0,27), PaxOnBoardIn (-0,46) and BagsIn (-0,54). On the other hand, TimeOfDay_midday correlates negatively with CargoIn (-0,27) but positively with PaxOnBoardOut (0,22) and BagsOut (0,33). TimeOfDay_evening has a negative correlation with BagsOut (-0,21), PaxOnBoardOut (-0,23), CargoIn (-0,3) and positive correlation with PaxOnBoardIn (0,43), BagsIn (0,46).

To summarize this trend related to transportation of people and material, emphasis of arriving cargo is mainly during mornings whereas the emphasis of departing passengers with their bags is during midday flights. Emphasis of arriving passengers and bags, on the other hand, is during the evenings.

There can be multiple reasons behind this phenomenon such as connecting flights from Helsinki but correlations related to this phenomenon are moderate at best. However, from turnaround point of view, this kind of information could be useful assuming that the placement of subprocesses on the critical path or the possible differences between loading and unloading of cargo and bags are known.

Lastly, the feature CargoIn correlates negatively with PaxOnBoardIn (-0,23) and BagsIn (-0,35). This indicates a relationship where the number of passengers and the amount of bags is less when arriving cargo is on board. The connection between features in this correlation is unclear and based solely on the correlation values, it is impossible to tell whether the small number of passengers and bags leads to higher amount of cargo being transported or vice versa. Whatever is the case, correlation between TurnaroundTime and CargoIn is close to negligible (0.12).

As discussed earlier, correlation only shows the linear association between variables, so it is not sensible to make highly developed conclusions based on the correlation coefficients. However, coefficients in Table 9 give some general information about the turnaround process. Based on correlation table it is possible to say that, on average, when the amount of arriving and departing passengers increases, turnaround time increases. Same applies also to deicing, which increases the likelihood of an increase in turnaround duration. However, deicing does not correlate highly with aircraft variant. Also, even though passengers and bags are drivers of turnaround time and the number of passengers and bags is partly dependent on the aircraft variant, aircraft variant itself does not considerably correlate with turnaround time. This could be interpreted so that attributes of a certain aircraft variant in general do not have such a big role in the determination of turnaround duration.

In addition, winter season has a positive impact on average to turnaround durations. Features MailIn, MailOut or FuelNeed do not indicate any considerable correlation with other features. Non-existent correlation provides insights into the turnaround process as it shows that there is no linear relationship between variables and therefore the usefulness of such features on the turnaround duration prediction might be non-existent as well. Turnaround has a weak correlation with flights occurring during midday but the reason behind such correlation should be examined before it could provide any operative benefits. Also, the behavior of people and material flows during different times of day could be useful but it requires more analysis on that topic so that the mechanisms behind that phenomenon could be revealed.

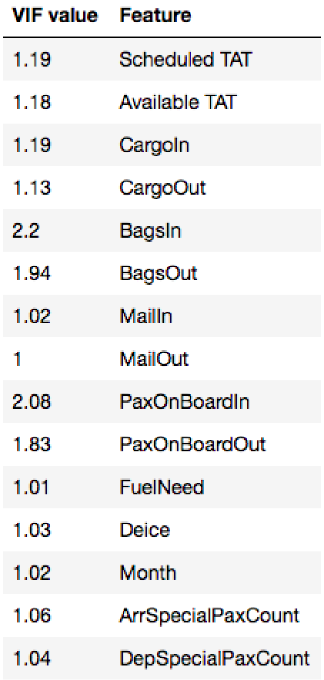
Estimations and guesses on the reasons behind correlations can be made but only further analysis can provide understanding of the causality between features. All in all, it is possible to state that no strong correlations exist in this data.

5.1.1 Multicollinearity

As mentioned in section 4.9 above, variance inflation factor (VIF) can be used in the determination of multicollinearity between variables. Table 10 below shows VIF values for different variables. It is worth noticing that dummy variables encoded from features ACGroup, Season and TimeOfDay are not included in the table as VIF produced for categorical variables does not provide sensible results.

According to Table 10, VIF is less than 5 for all the variables, and as discussed earlier, VIF value of 1 indicates no collinearity and value between 1 and 5 indicates moderate collinearity. This means that only weak or non-existent collinearity exists between given variables except for variables BagsIn, BagsOut, PaxOnBoardIn and PaxOnBoardOut where VIF value is close to 2. These findings are in line with the previous findings from the correlation table.

Table 10: Variance inflation factor values for different variables



VIF value	Feature
1.19	Scheduled TAT
1.18	Available TAT
1.19	CargoIn
1.13	CargoOut
2.2	BagsIn
1.94	BagsOut
1.02	MailIn
1	MailOut
2.08	PaxOnBoardIn
1.83	PaxOnBoardOut
1.01	FuelNeed
1.03	Deice
1.02	Month
1.06	ArrSpecialPaxCount
1.04	DepSpecialPaxCount

As correlations can only provide one point of view into the inspection of relationships between variables, it is used together with interaction values to build a bigger picture of causalities in an aircraft turnaround. Next, bivariate visualizations are reviewed.

5.2 Exploratory data analysis through bivariate visualizations

In this section a set of plots are displayed to provide exploratory data analysis and to visualize findings from the correlation table. Figure 14 below shows the relationship between average monthly turnaround time specified for different aircraft variants. Average turnaround times are highest during winter months according to Figure 14 and start to decrease towards summer months. However, average monthly turnaround times start to increase again towards July. Reason for this is unclear but Figure 15 indicates the average monthly turnaround time per passenger (arriving + departing) specified for different aircraft variants. It shows that average turnaround decreases from winter months towards the summer months. This means that the increase of average turnaround times during winter months is not caused by the passenger numbers. According to Figure 14, time difference for average turnaround times between different aircraft variants is a maximum of over 3 minutes which is during July. One interesting note from Figure 14 and Figure 15 is that on average A321 has longer turnarounds compared to A320 and A319 but on average the turnaround time per passenger is lowest for A321 compared to the other two.

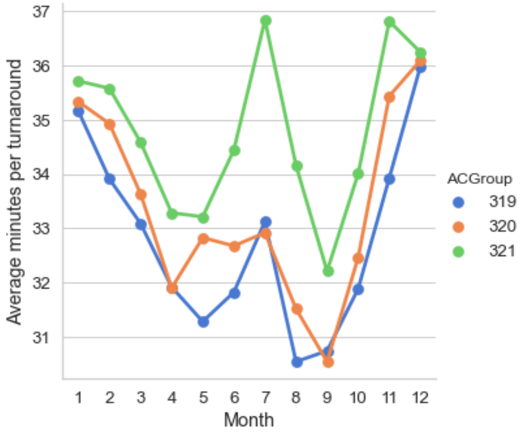


Figure 14: Average monthly turnaround time for different aircraft variants

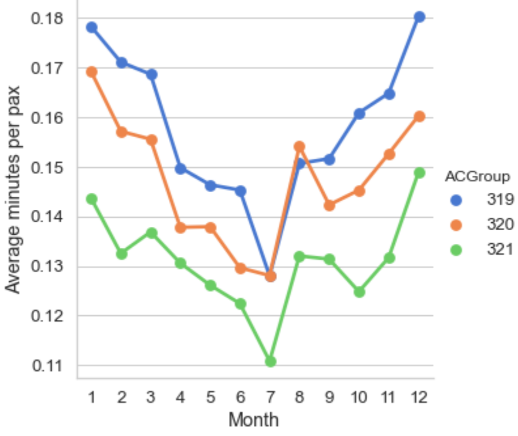


Figure 15: Average monthly turnaround time per passenger for different aircraft variants

Figure 16 furthermore demonstrates the difference between the average monthly turnaround per passenger for different aircraft variants with a distinction between flights with and without deicing. Reddish lines indicate turnarounds where the deicing process has been performed whereas blue lines indicate turnaround without deicing. Figure points out that on average, deicing brings an increase to a turnaround but in addition figure demonstrates that flights without the deicing process have a higher average turnaround per passenger during winter months. This implies that during winter months there are other factors that prolong turnarounds.

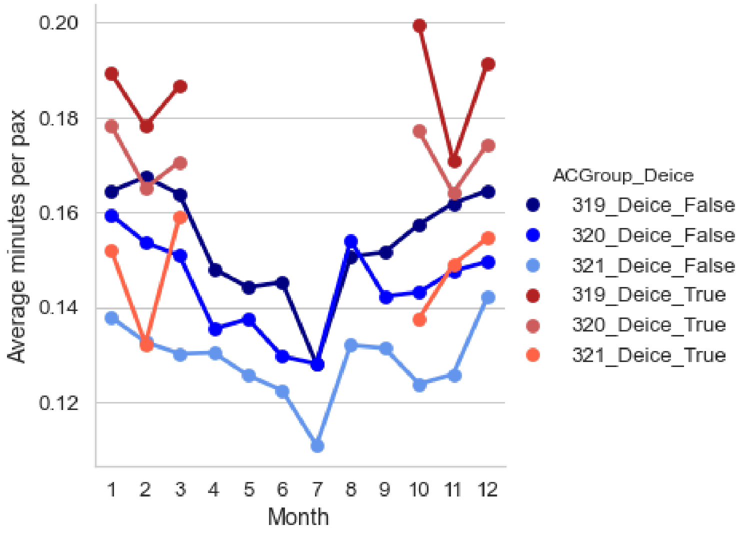


Figure 16: Average monthly turnaround time per passenger for different aircraft variants with and without deicing

Boxplots in Figure 17 indicate that not only the turnarounds are longer during winter months with a median of 35 minutes compared to approximately 32 minutes during other seasons, but also the TAT difference has a median value of almost 5 minutes for winter whereas the median of TAT difference for other seasons is zero minutes.

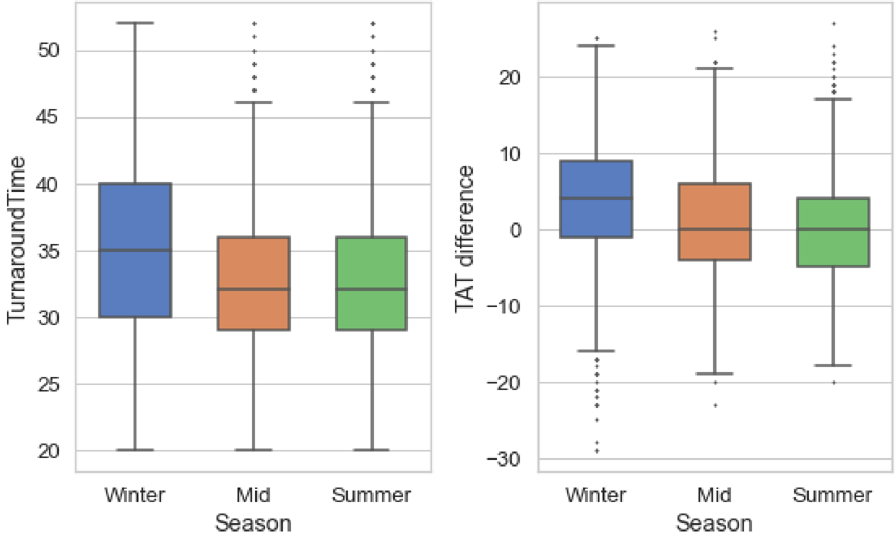


Figure 17: Turnaround time and TAT difference for different seasons

Scatterplots in Figure 18 visualize the mean turnaround times for arriving and departing passenger counts. From figures below, it can be concluded that the increase in the number of departing passengers, that is, the passengers that board the plane in airport A, have a larger impact on the turnaround time compared to the increase in the number of arriving passengers. This indication is in line with the findings from correlation table where feature PaxOnBoardOut had a larger correlation with TurnaroundTime compared to the feature PaxOnBoardIn. It is worth to note that the trend for the effect on the turnaround time due to increase in number of passengers is almost identical between aircraft variants.

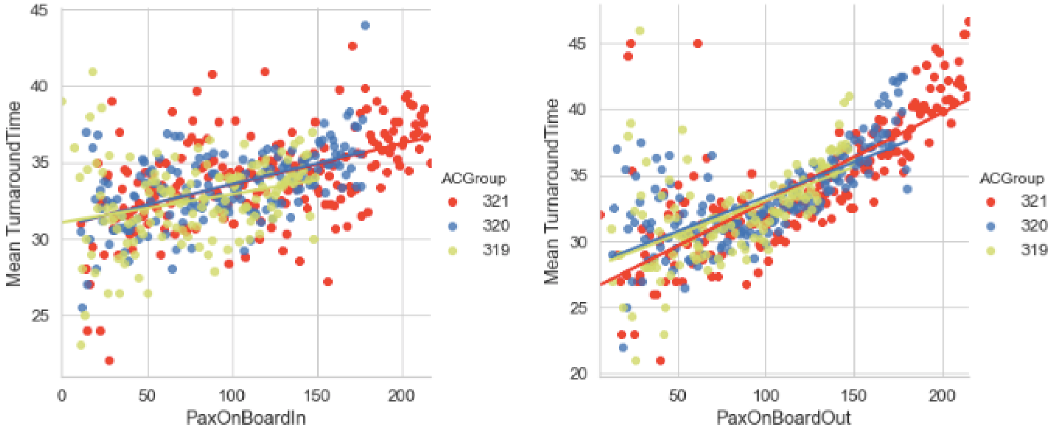


Figure 18: Arriving and departing passenger count versus mean turnaround time

According to scatterplots in Figure 19 below, the relationship between turnaround time and the amount of arriving and departing bags is similar to the relationship between turnaround time and passenger count. In this case for outgoing bags the linear trend line for A321 is a bit steeper compared to other aircraft variants.

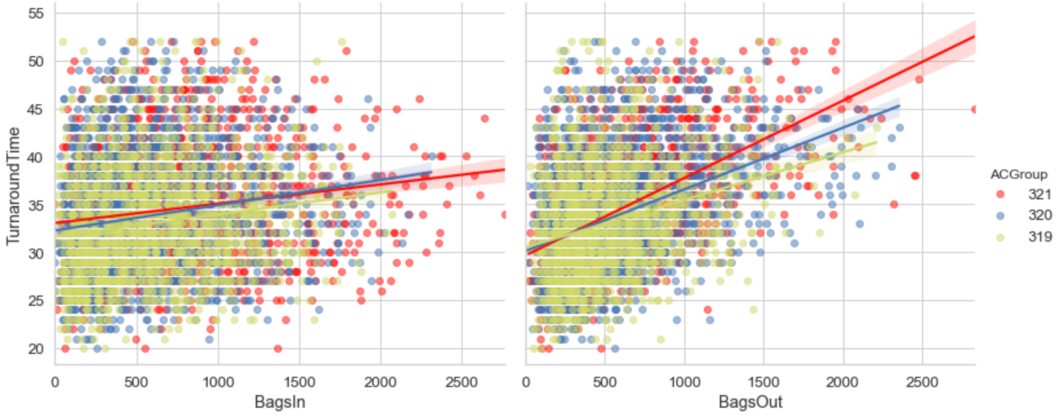


Figure 19: Amount of arriving and departing bags versus turnaround time

Barplots below in Figure 20 show the changing number of arriving and departing passengers on different times of a day. Due to the sensitiveness of information related to passenger load factors, turnarounds have been divided into two groups of equal size with other group presenting the turnarounds with smaller and the other one the turnarounds with larger half of passengers. From the figures it can be concluded that largest number of passengers arrives during evenings whereas passenger counts are smallest during mornings. For the number of departing passengers the difference is not as large between different times of day compared to the number of arriving passengers. However, from the figure it can be noted that larger number of passengers departs during middays and smallest number departs during the mornings.

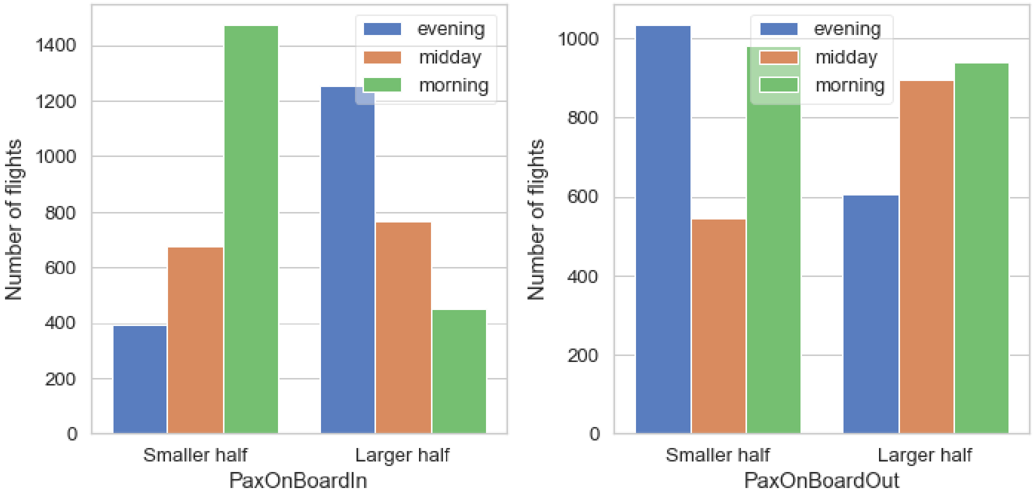


Figure 20: Arriving and departing number of passengers in quantiles for different times of day

First boxplot in Figure 21 below shows the relationship between turnaround time and the feature TimeOfDay whereas the second boxplot displays relationship between TAT diff and TimeOfDay. Based on the correlations in Table 9, turnaround time had highest correlation with the feature PaxOnBoardOut. By comparing the observation from Figure 20 above we can see that number of departing passengers is highest during midday and also turnaround time and TAT difference are highest during midday. Turnaround times during midday have a mean of 36 minutes which is 4 minutes higher than the mean values for turnaround during mornings (32 minutes) and 2 minutes higher as the mean value for evenings (34 minutes). Also, TAT difference boxplot demonstrates that major part of flights during middays have longer actual turnaround times than scheduled times for those flights. For this dataset, these findings indicate that on average turnarounds have longest durations on middays and one factor behind this phenomenon is on average the large number of departing passengers.

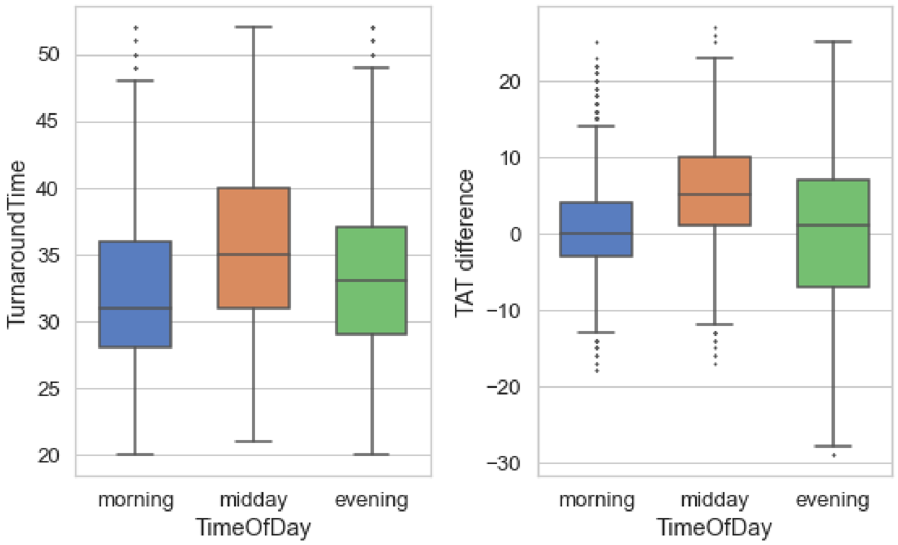


Figure 21: Turnaround time and TAT difference for different times of day

Features BagsIn and BagsOut had similar correlations with TimeOfDay features as features PaxOnBoardIn and PaxOnBoardOut. That observation can be verified from Boxplots in Figure 22.

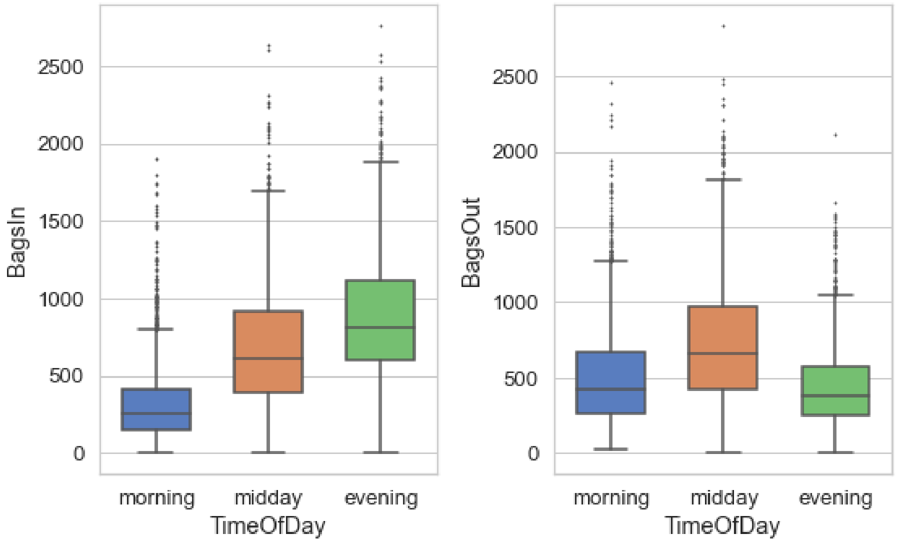


Figure 22: Amount of arriving and departing bags for different times of day

Figure 23 displays the correlation between turnaround time and deicing. From this figure it is possible to note that the minimum duration for a turnaround with deicing is 25 minutes whereas minimum time for turnarounds without deicing is 20 minutes. On the other hand, the difference between maximum duration for turnarounds with and without deicing is not as strong even though there are more observations at the top of the plot for turnarounds with deicing. Turnarounds with deicing have, on average, longer durations and the mean value for TAT diff when deicing is performed is 7.6 minutes whereas the mean TAT diff for turnarounds without deicing is 0.5 minutes.

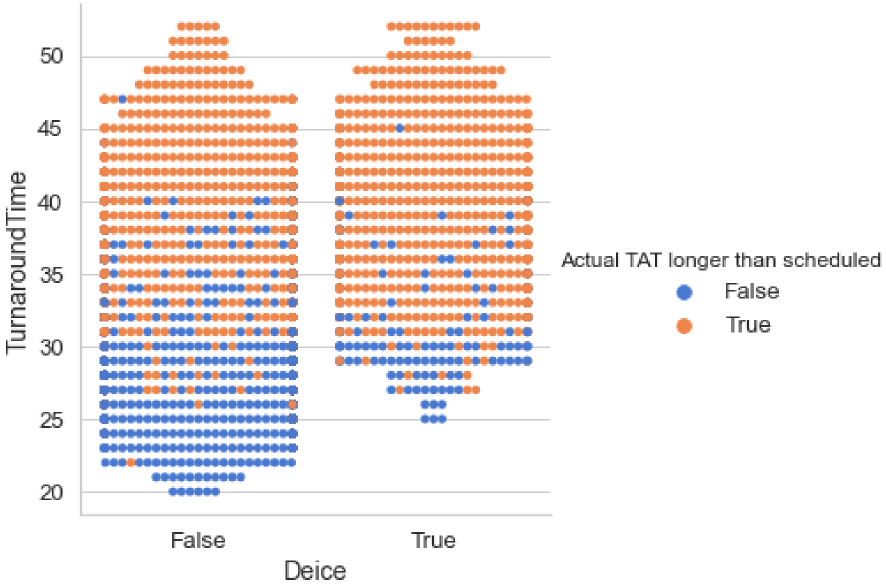


Figure 23: The effect of deice on turnaround times

Based on the dataset, inbound cargo arrives for 54 percent of cases during mornings, 23 percent of cases during middays and 23 percent of cases during evenings. On the other hand, outbound cargo leaves airport A most typically during middays which equals to 46 percent of cases and less likely during mornings and evenings which represent 22 percent and 33 percent of cases, respectively.

If this observation is compared with the information from Figure 21, it seems that during middays both the largest amount of cargo and largest number of passengers with their bags are most likely to depart. On the other hand, during mornings, a smaller number of passengers but the largest amount of inbound cargo arrives on average.

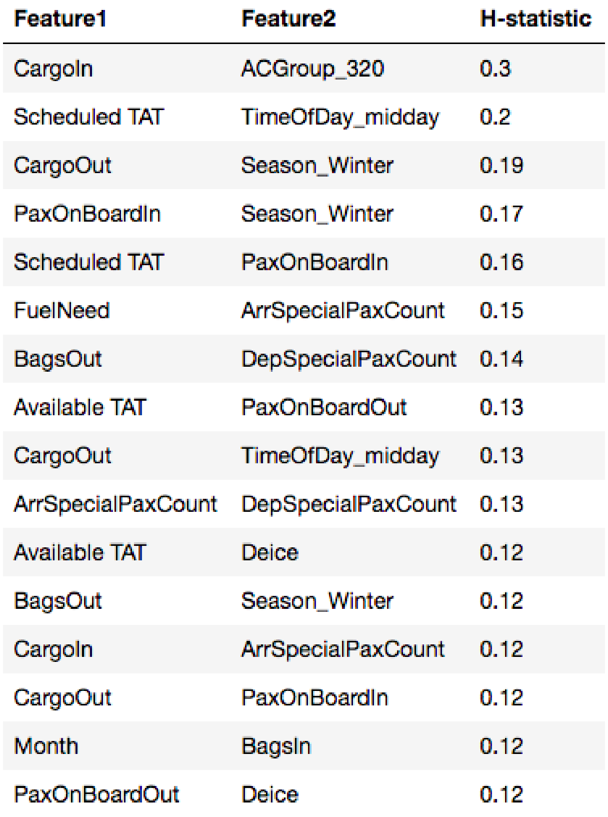
5.3 Interaction with H-statistic

As presented in the earlier chapter, H-statistic is one of the methods that are used to inspect potential interactions between variables. Also SHAP interaction values are used in the analysis of interactions but they will be calculated by using the fully trained XGBoost model. Calculation of H-statistic has been done by using Python framework sklearn-gbmi (Haygood, 2020), which uses a gradient-boosting model.

Below in Table 11 are presented the H-statistic values for different variable pairs. Based on Molnar (2019), it is hard to determine when a value of the H-statistic can be considered strong and how much the value needs to be above 0 to be considered significant. By bearing in mind that H-statistic value 0 indicates no interaction and value of 1 implies that all the effect comes from the interaction, a threshold of 0.10 has been selected to filter out less promising candidates for Table 11.

A visual inspection was perfomed for variable pairs in the table by using a partial dependence plot and only plots for feature pairs CargoOut-Season, PaxOnBoardIn-Season, BagsOut-DepSpecialPaxCount as well as BagsOut-Season are presented in this section. The reason for not including all the remaining pairs in the table is that either the interaction will be presented in the section displaying SHAP or that the interaction dispayed inconsistencies, was not backed by data or was not significant enough. Last was the case for partial dependence plot drawn for feature pairs CargoIn and ACGroup, which indicated that interaction effects are present between aircraft variants with a maximum difference of 0.18 minutes in the turnaround time.

Table 11: Interactions for pairs of variables with H-statistic above 0.10



Feature1	Feature2	H-statistic
CargoIn	ACGroup_320	0.3
Scheduled TAT	TimeOfDay_midday	0.2
CargoOut	Season_Winter	0.19
PaxOnBoardIn	Season_Winter	0.17
Scheduled TAT	PaxOnBoardIn	0.16
FuelNeed	ArrSpecialPaxCount	0.15
BagsOut	DepSpecialPaxCount	0.14
Available TAT	PaxOnBoardOut	0.13
CargoOut	TimeOfDay_midday	0.13
ArrSpecialPaxCount	DepSpecialPaxCount	0.13
Available TAT	Deice	0.12
BagsOut	Season_Winter	0.12
CargoIn	ArrSpecialPaxCount	0.12
CargoOut	PaxOnBoardIn	0.12
Month	BagsIn	0.12
PaxOnBoardOut	Deice	0.12

Figure 24 below displays the interaction effects between variables CargoOut and Season which have a H-statistic of 0.19. According to the figure, during winter season the variable CargoOut has a larger effect on turnaround duration the greater is the value of the CargoOut variable. For other seasons and values of CargoOut variable, there is no similar interaction effect present.

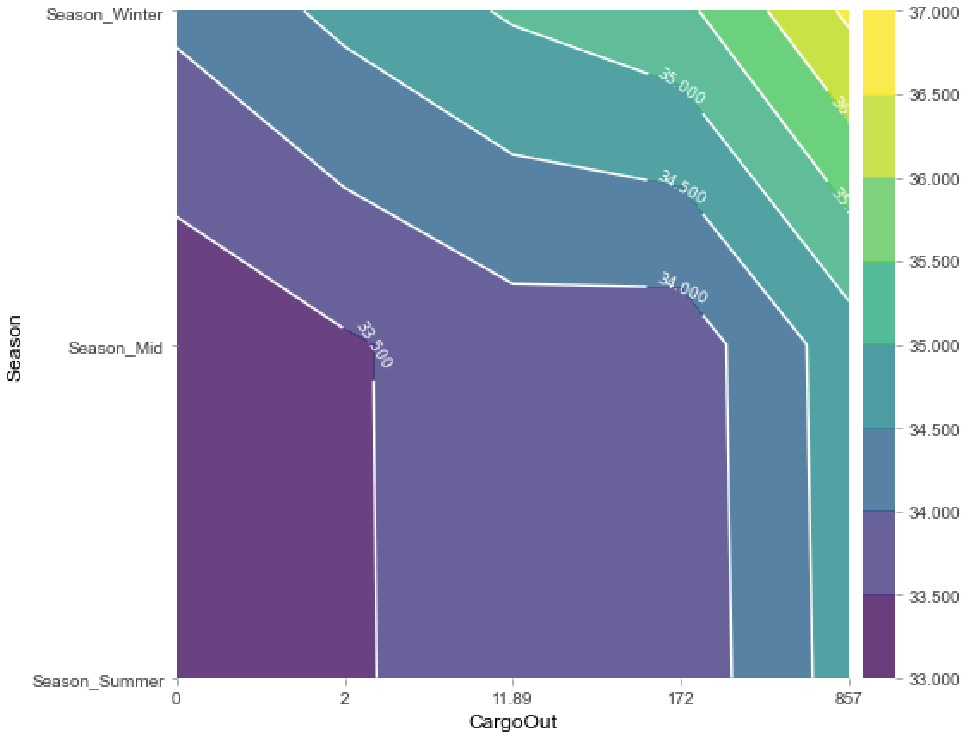


Figure 24: Partial dependence plot for variables CargoOut and Season

Figure 25 below for features PaxOnBoardIn and Season, on the other hand, demonstrates that during Season_Winter turnarounds are longer for turnarounds with a large number of arriving passengers as compared to other seasons. H-statistic for this variable pair is 0.17 and figure indicates that highest turnaround durations occur when number of arriving passengers is approximately 170. Based on Table 11, variable pair Month and BagsIn demonstrate similar interaction even though H-statistic has a lower value of 0.12 for them.

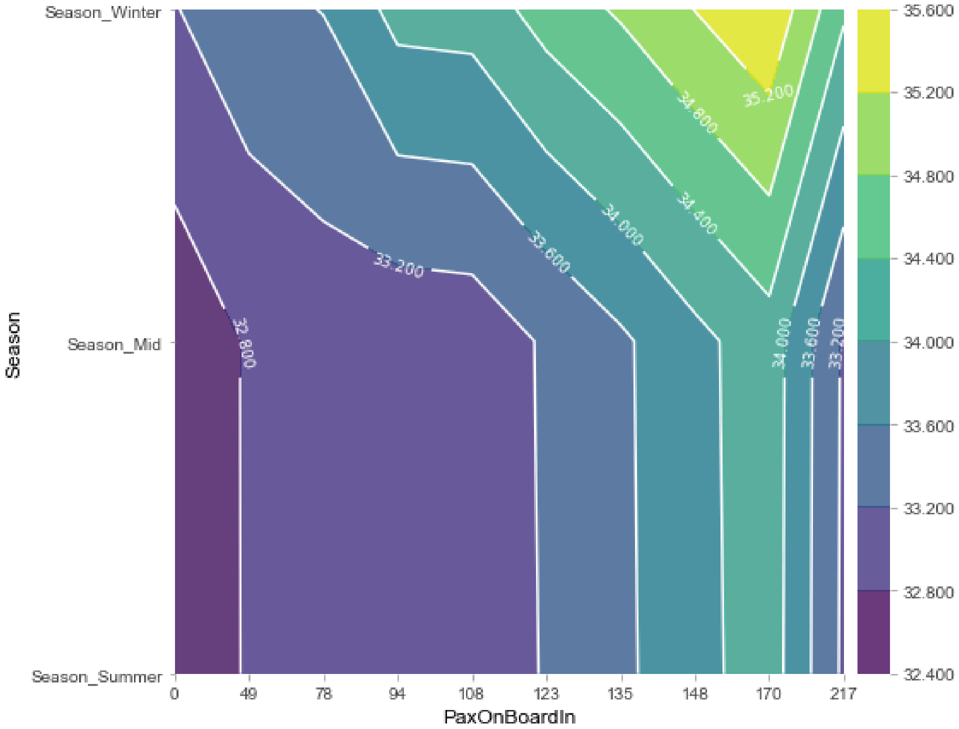


Figure 25: Partial dependence plot for variables PaxOnBoardIn and Season

Figure 26 below displays that for turnarounds where there is simultaneously large amount of departing bags and at least one departing special passenger, the expected turnaround duration is higher compared to a situation when only either of these variables has a high value. The reason for this observation is not evident and there is no similar connection between high number of departing special passengers and high number of departing passengers even though passengers and bags have a correlation.

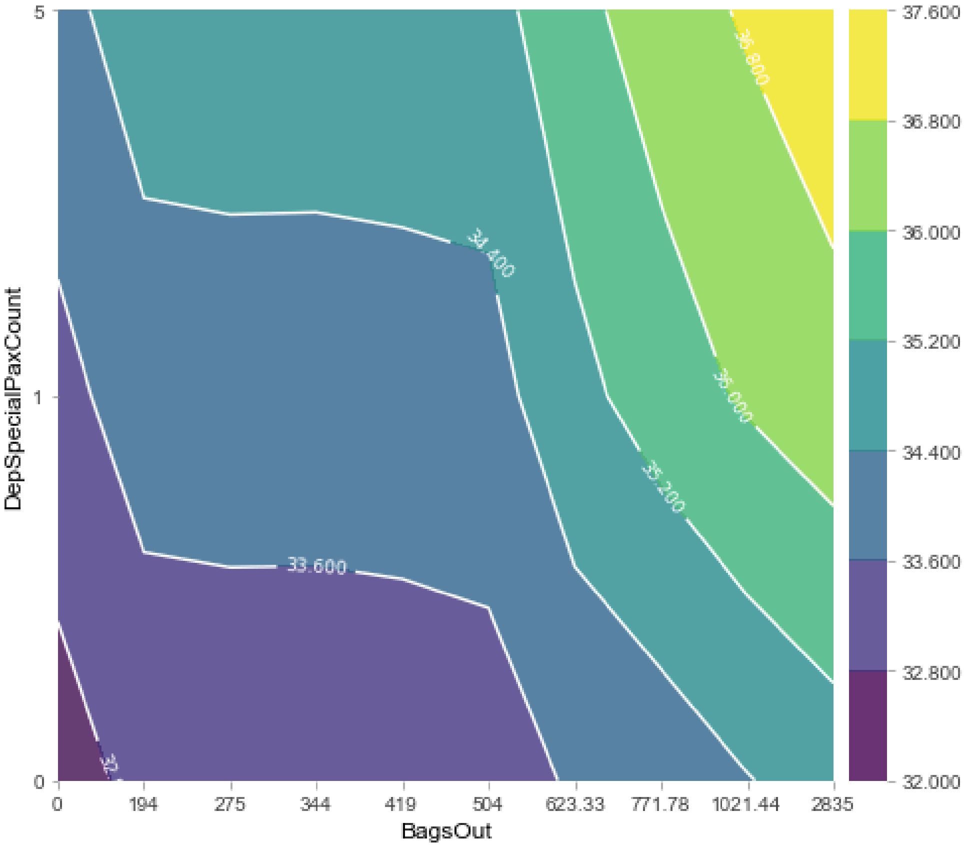


Figure 26: Partial dependence plot for variables BagsOut and DepSpecialPaxCount

Lastly, Figure 27 below indicates that during winter season turnarounds for a certain amount of departing bags are longer compared to other seasons with the same amount of bags. Similar interaction is not present between features PaxOnBoardOut and Season. It is worth to note, that Season has a similar interaction with features BagsOut and CargoOut which could indicate that a mechanism exists which has an effect that slows down the loading process of cargo and bags during winter season.

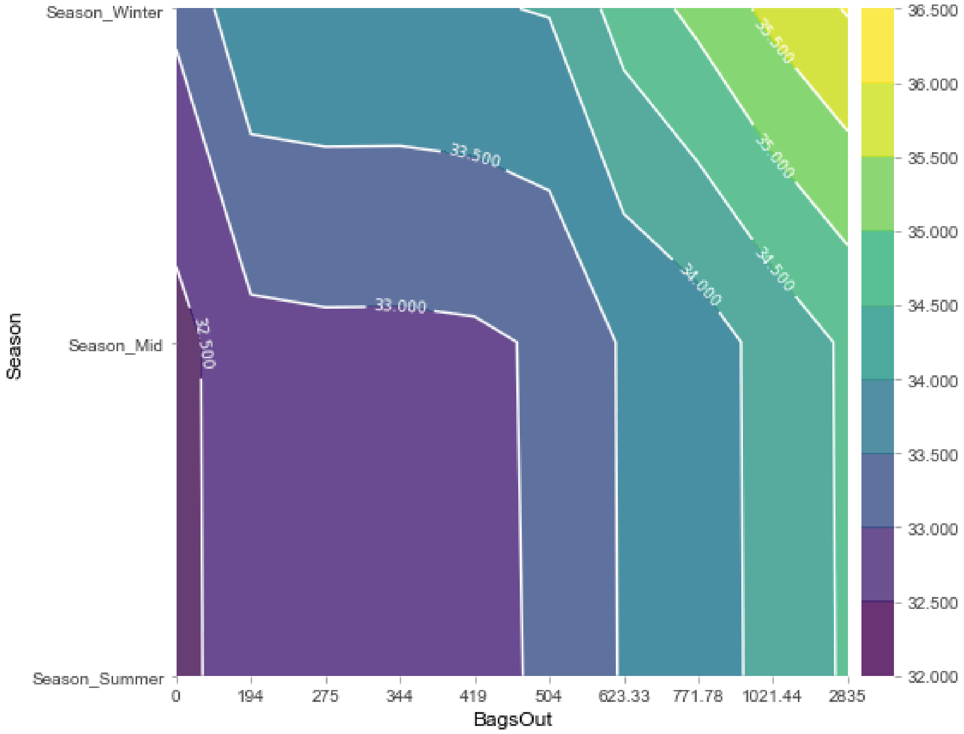


Figure 27: Partial dependence plot for variables BagsOut and Season

Figures above have been created by using the PDPbox Python library (chakki, 2020). Partial dependence plots aid in the visualization of interactions but they should be considered mainly as a visual aid which can help in the determination of the trend of the interaction. Table 11 indicates that feature pair FuelNeed and ArrSpecialPaxCount have an interaction of strength 0.15. However, the visualization presented below in Figure 28 shows that turnarounds where fuelling is performed the turnaround has a longer duration without arriving special passengers compared to the case with 5 special passengers. In reality, the dataset does not contain a turnaround which included both fuelling and 5 arriving special passengers. In addition, logical reasoning would say that turnarounds with larger numer of arriving special passengers would take a longer time to be performed. In the next section, the feature selection process will be presented.

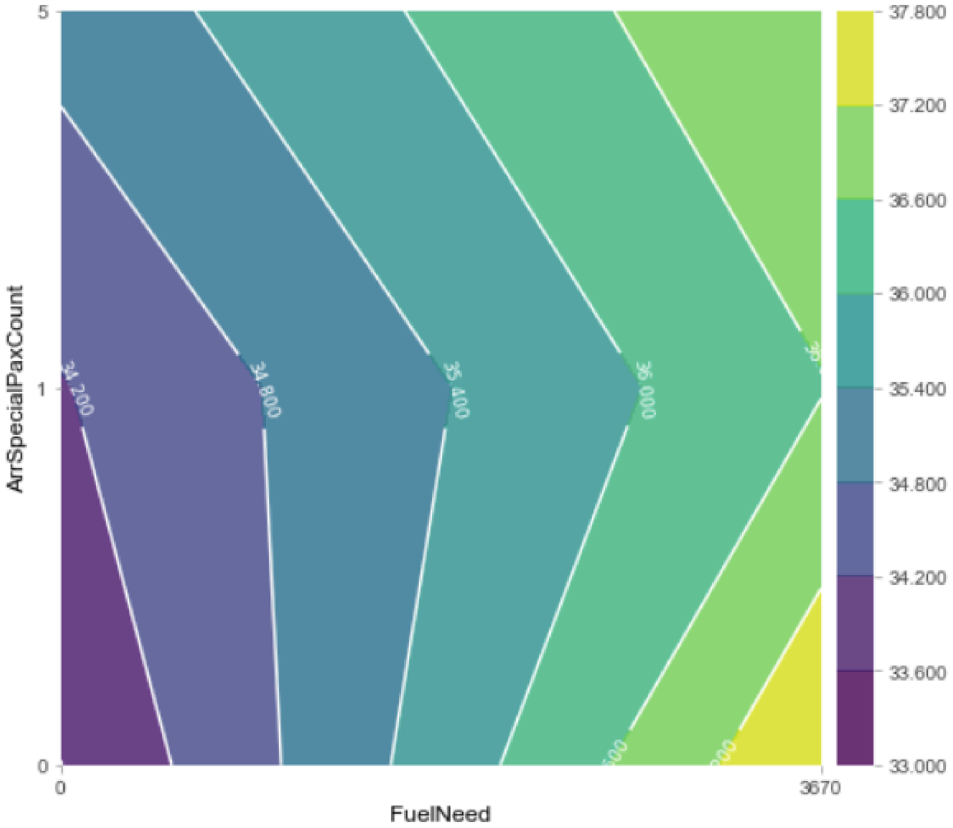


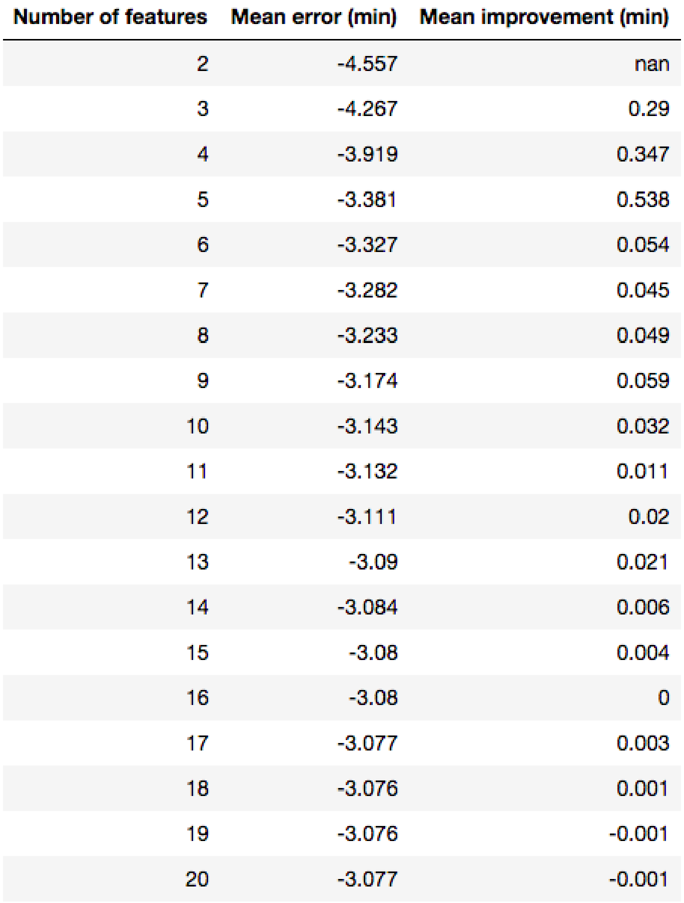
Figure 28: Partial dependence plot for variables FuelNeed and ArrSpecialPaxCount

5.4 Feature selection

Four sets of features are selected for the machine learning algorithm XGBoost. First includes the full set of features which is eventually also provided for the SHAP framework. Second and third feature sets include only a subset of features and they have been selected based on the RFE method. Second subset is provided by the RFE using a random forest algorithm and the third subset, on the other hand, is provided by using XGBoost with default hyperparameter values. Fourth subset has all the features except Available TAT and Scheduled TAT.

Before the determination of essential features for second and third subset, the determination of optimal number of features in general was needed to be made. Table 12 provides the average negative MAE value for different subset sizes and the average improvement in accuracy from including an additional feature into the subset. Figure 29 visualizes the difference of error values for different subset sizes. Optimal number of features were selected by setting the threshold for improvement to 0.05 minutes. This means that the last set for which addition of a new feature provided more than 0.05 minutes gain in accuracy was selected as the optimal subset size which corresponds to subset size of 9 features. Figure 29 visualizes the improvement by showing that after fifth feature the improvement in accuracy decreases significantly and after ninth feature the improvement decreases to very small.

Table 12: MAE for different feature subset sizes



Number of features	Mean error (min)	Mean improvement (min)
2	-4.557	nan
3	-4.267	0.29
4	-3.919	0.347
5	-3.381	0.538
6	-3.327	0.054
7	-3.282	0.045
8	-3.233	0.049
9	-3.174	0.059
10	-3.143	0.032
11	-3.132	0.011
12	-3.111	0.02
13	-3.09	0.021
14	-3.084	0.006
15	-3.08	0.004
16	-3.08	0
17	-3.077	0.003
18	-3.076	0.001
19	-3.076	-0.001
20	-3.077	-0.001

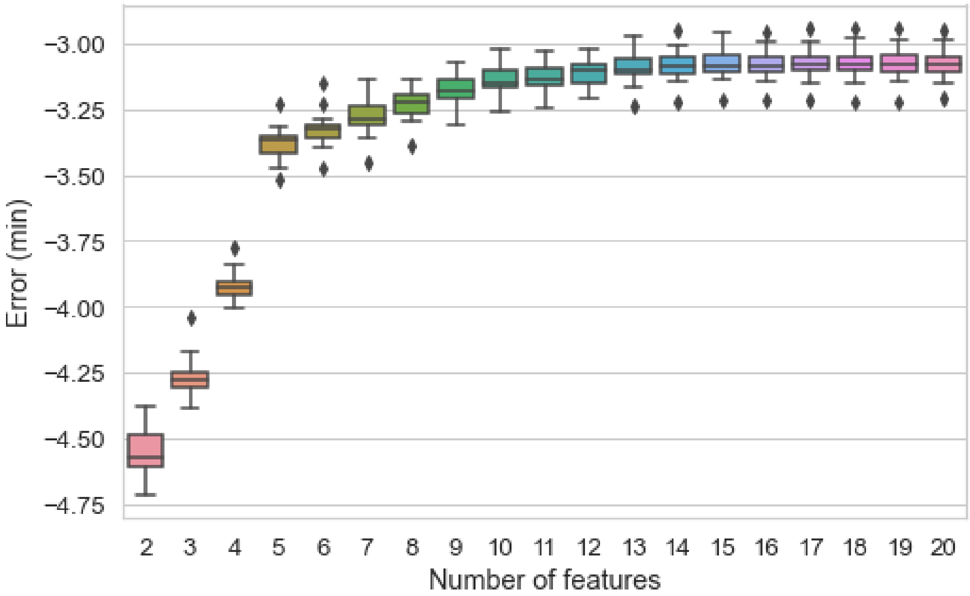


Figure 29: Visualization of MAE for different feature subset sizes

After determination of optimal number of features, RFE was used to provide the selected features for that specific subset size. Table 13 and Table 14 below show the ranking made by the RFE for different features. Rank number 1 means that it was included in the subset. Tables display that the two subsets produced by different algorithms are similar but have also some differences. Features that are included in both subsets include Available TAT, PaxOnBoardOut, BagsOut, Deice, TimeOfDay_midday and Month. Differences between subsets are features CargoOut, BagsIn and PaxOnBoardIn which are included in the set provided by the random forest whereas features FuelNeed, DepSpecialPaxCount and TimeOfDay_morning are in the subset chosen by non-tuned XGBoost.

Table 13: Feature ranking with random forest

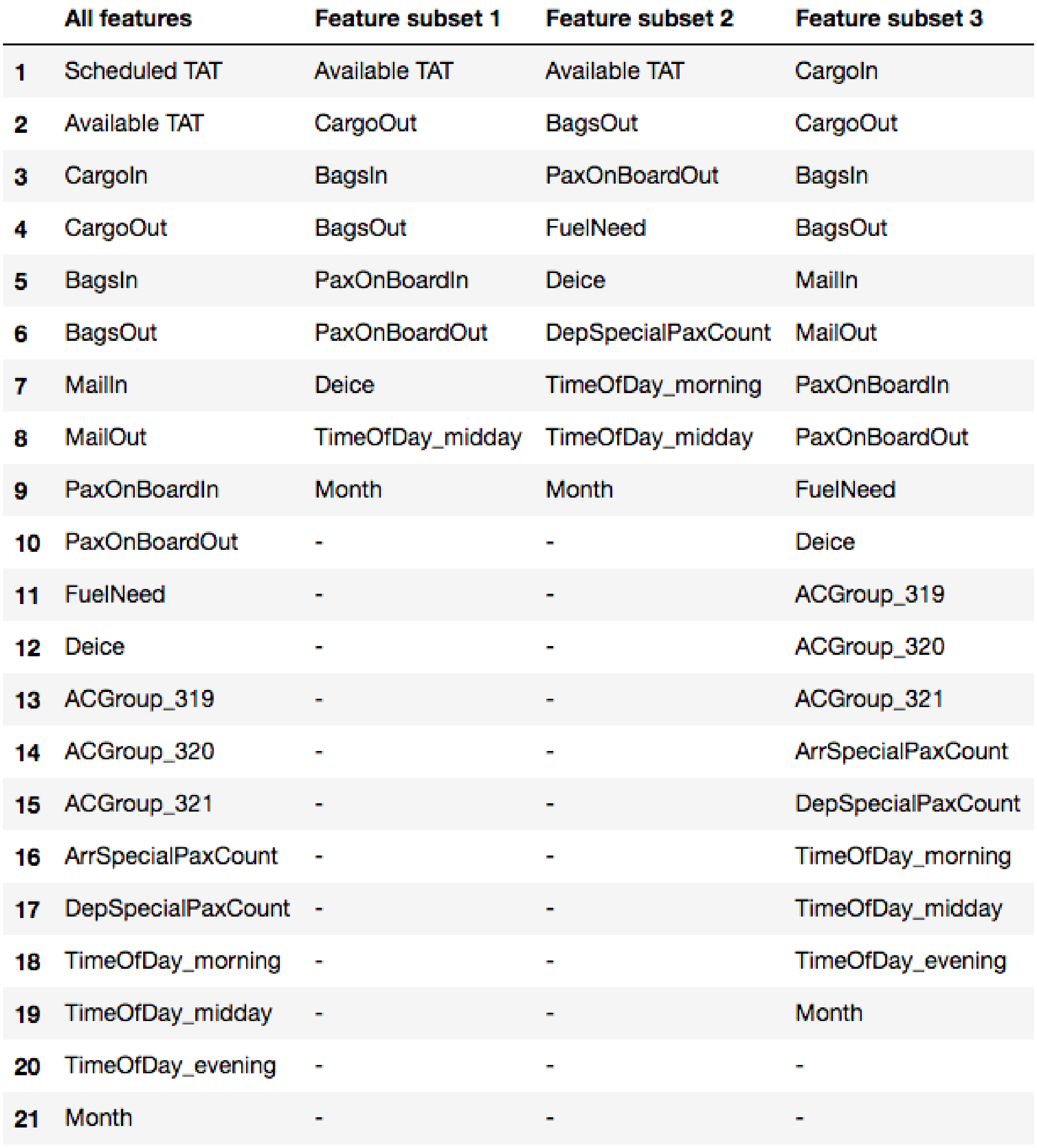
Feature	Ranking by random forest
Month	1
Available TAT	1
CargoOut	1
BagsIn	1
BagsOut	1
TimeOfDay_midday	1
Deice	1
PaxOnBoardIn	1
PaxOnBoardOut	1
CargoIn	2
Scheduled TAT	3
DepSpecialPaxCount	4
TimeOfDay_morning	5
ArrSpecialPaxCount	6
ACGroup_320	7
ACGroup_321	8
ACGroup_319	9
FuelNeed	10
TimeOfDay_evening	11
MailIn	12
MailOut	13

Table 14: Feature ranking with XGBoost

Feature	Ranking by XGBoost
FuelNeed	1
TimeOfDay_midday	1
TimeOfDay_morning	1
DepSpecialPaxCount	1
Deice	1
PaxOnBoardOut	1
Month	1
BagsOut	1
Available TAT	1
TimeOfDay_evening	2
CargoOut	3
PaxOnBoardIn	4
ArrSpecialPaxCount	5
ACGroup_320	6
ACGroup_321	7
ACGroup_319	8
BagsIn	9
MailIn	10
CargoIn	11
Scheduled TAT	12
MailOut	13

Features PaxOnBoardOut, BagsOut, Deice, Month and TimeOfDay_midday which are present in both subsets and their relations to the target variable TurnaroundTime have been discussed already. However, feature Available TAT and its connection to other variables and turnaround time is unclear based on the information provided so far. Summary of different subsets and their features which are provided for the XGBoost algorithm are shown in Table 15 below. In the next section, results of the trained XGBoost model will be reviewed.

Table 15: Summary of different feature subsets



	All features	Feature subset 1	Feature subset 2	Feature subset 3
1	Scheduled TAT	Available TAT	Available TAT	CargoIn
2	Available TAT	CargoOut	BagsOut	CargoOut
3	CargoIn	BagsIn	PaxOnBoardOut	BagsIn
4	CargoOut	BagsOut	FuelNeed	BagsOut
5	BagsIn	PaxOnBoardIn	Deice	MailIn
6	BagsOut	PaxOnBoardOut	DepSpecialPaxCount	MailOut
7	MailIn	Deice	TimeOfDay_morning	PaxOnBoardIn
8	MailOut	TimeOfDay_midday	TimeOfDay_midday	PaxOnBoardOut
9	PaxOnBoardIn	Month	Month	FuelNeed
10	PaxOnBoardOut	-	-	Deice
11	FuelNeed	-	-	ACGroup_319
12	Deice	-	-	ACGroup_320
13	ACGroup_319	-	-	ACGroup_321
14	ACGroup_320	-	-	ArrSpecialPaxCount
15	ACGroup_321	-	-	DepSpecialPaxCount
16	ArrSpecialPaxCount	-	-	TimeOfDay_morning
17	DepSpecialPaxCount	-	-	TimeOfDay_midday
18	TimeOfDay_morning	-	-	TimeOfDay_evening
19	TimeOfDay_midday	-	-	Month
20	TimeOfDay_evening	-	-	-
21	Month	-	-	-

5.5 XGBoost model training and assessment

This section gives an assessment on the results of the trained XGBoost model. As mentioned before, four different models have been trained and each model is provided with its own set of features. Training was conducted with nested cross-validation. In the nested cross-validation a randomized search was used to pick different hyperparameter value combinations from a parameter grid and the predictive performance for those values was calculated. Inner and outer loops of the nested cross-validation consisted of five folds. In addition, 30 iterations were performed for the selection of different hyperparameter combinations in the inner loop. Eventually, for each feature set nested cross-validation returned a model with best performing hyperparameters in terms of test error.

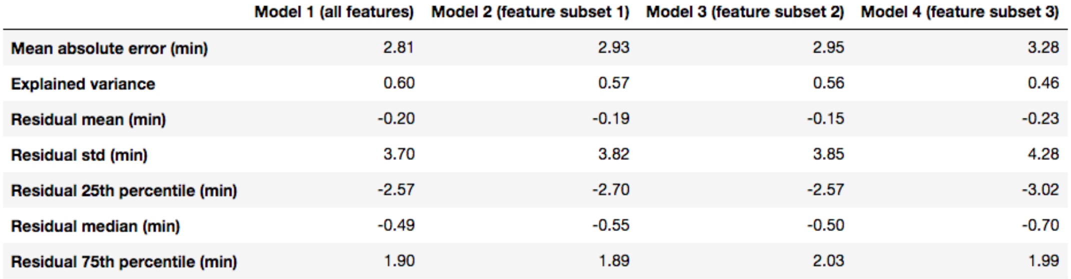
In addition to the data given for the model in the training phase, 10 percent of the original data was provided for the fully trained models as previously unseen test data. This batch of unseen data was used both in the performance assessment of the models as well as in the analysis made with SHAP values. SHAP values were calculated by the model with all features.

Table 16 below displays the mean absolute error (MAE) and explained variance for all the four models. Information regarding the distributions of the residuals, that is, the differences between actual and predicted turnaround times are provided in the same table. Best performing MAE of 2.81 minutes is for model 1 which was provided with all the features. Explained variance for model 1 is 0.60 which means that the predictions were able to explain approximately 60 percent of the variation in the actual turnaround durations. For the models 2 and 3 the error metric MAE is not significantly lower compared to model 1. Also, the explained variance for these models is close to the best performing model 1. Weakest performing model is model 4 which is provided with a feature subset that did not contain Available TAT or Scheduled TAT with MAE of 3.28 minutes and explained variance of 0.46.

The difference for explained variance between models 1 and 4 is 0.14. This indicates that information related to the scheduled turnaround duration in the input data, which is ultimately based on history data and data provided by the manufacturer, brings an increase of 14 percentage points to the explained variance. Therefore, from a turnaround modelling perspective, a model with varying turnaround related input parameters without prior knowledge related to turnaround durations is capable of explaining only the said 46 percent of variations in turnaround durations.

Models 2 and 3 reach a mean error metric of 2.93 minutes and 2.95 minutes with explained variances of 0.57 and 0.56 which are close to the best mean error of 2.81 minutes and best explained variance of 0.60. Common features in subsets 1 and 2 are Available TAT, PaxOnBoardOut, BagsOut, Deice, Month and TimeOfDay_midday as discussed in the previous section. It means that these features are capable of explaining major part of the predictions. Same features demonstrated their connection to turnaround time through correlations. Explained variance of 0.60 for Model 1 with full set of features means that additional features are required to explain the missing 40 percent of variance and might be unexplainable by the current dataset. Information in Table 16 regarding the distributions of residuals is visualized in Figure 30 below.

Table 16: Metrics for different XGBoost models



	Model 1 (all features)	Model 2 (feature subset 1)	Model 3 (feature subset 2)	Model 4 (feature subset 3)
Mean absolute error (min)	2.81	2.93	2.95	3.28
Explained variance	0.60	0.57	0.56	0.46
Residual mean (min)	-0.20	-0.19	-0.15	-0.23
Residual std (min)	3.70	3.82	3.85	4.28
Residual 25th percentile (min)	-2.57	-2.70	-2.57	-3.02
Residual median (min)	-0.49	-0.55	-0.50	-0.70
Residual 75th percentile (min)	1.90	1.89	2.03	1.99

Figure 30 visualizes the kernel density estimations of residuals for the four models. By combining information from Table 16 and Figure 30 it can be noted that distribution of model 1 residuals has the largest density around the residual value of approximately -0.49 minutes with standard deviation of 3.70 minutes. Model 4, which has the worst performance, has also the largest dispersion of residuals with a standard deviation of 4.28 minutes and median of -0.70 minutes.

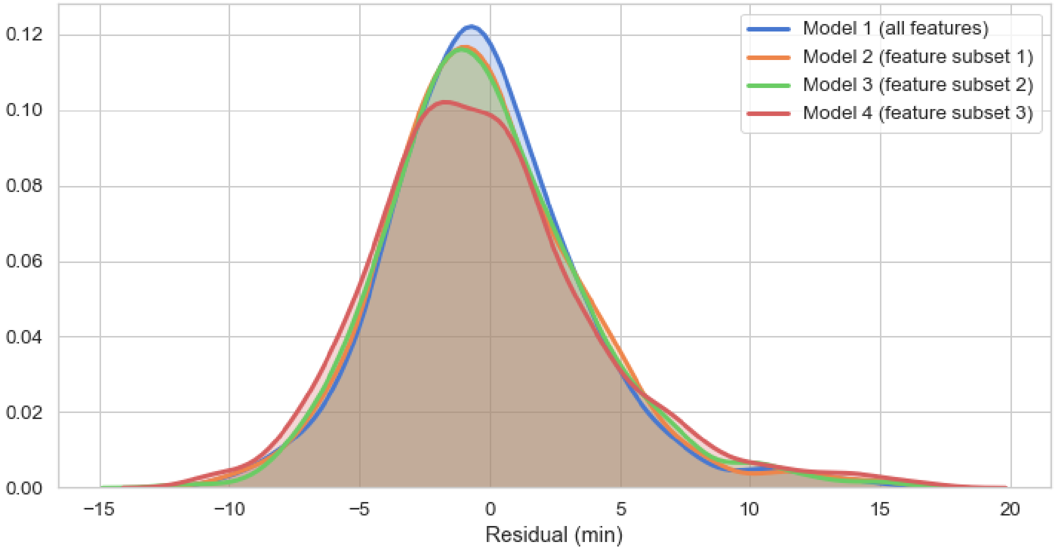


Figure 30: Distributions of residuals for different XGBoost models

All models have a negative mean for residuals which indicates that, on average, models predict a larger turnaround duration compared to the actual turnaround duration. In addition, all distributions in Figure 30 have a tail on the right side which indicates that most extreme errors between actual values and predictions happen when models underestimate the duration of a turnaround. This is reasonable, as in practice in a situation when the length of a turnaround is optimized to be as short as possible, it is presumably more difficult to be multiple minutes ahead of schedule during turnarounds compared to being late. If the reasons for delay are due to reasons that the dataset is not able to justify, consequently the machine learning is not capable of predicting such delays either.

One benchmark to assess the operational usefulness of the model is to compare the difference between the actual and scheduled turnaround duration with the difference between the actual and predicted turnaround duration. Such comparison demonstrates in practice how well the predictions have determined the actual turnaround duration compared to the scheduled turnaround time, which in some cases might be the only estimate of a coming turnaround duration. Figure 31 below illustrates the distribution of this difference between the absolute values of these two residuals mentioned. For 75 percent of turnarounds in the test set the predictions have had a smaller error in estimating the turnaround compared to the error between actual and scheduled turnaround durations.

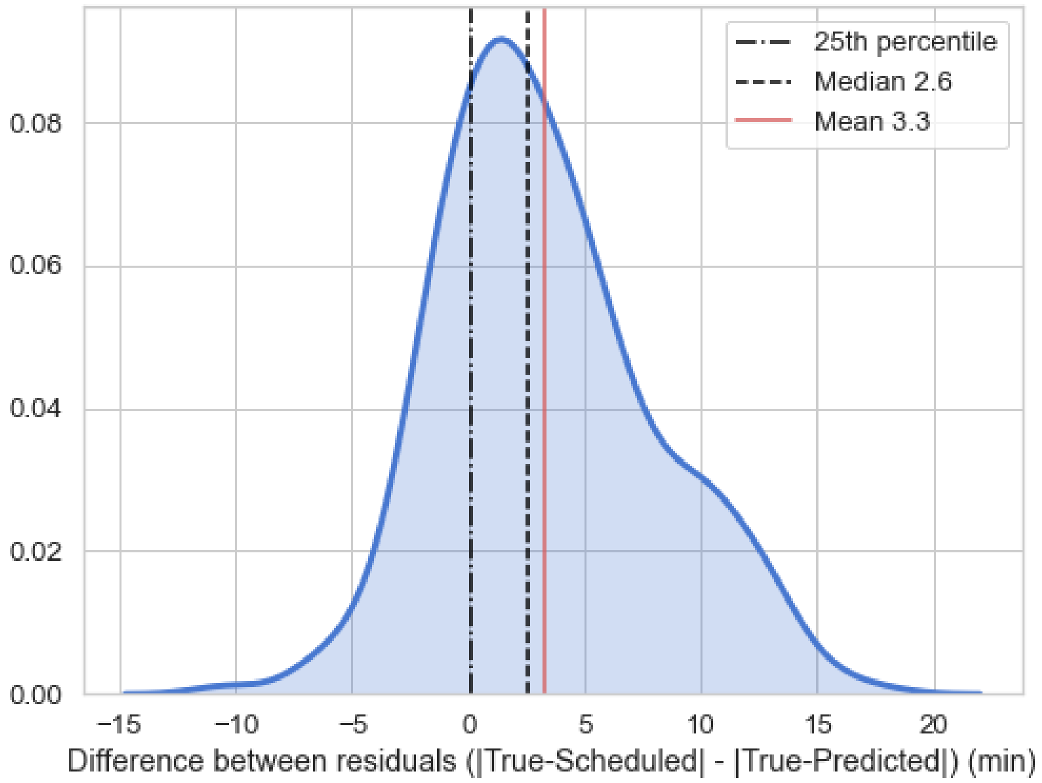
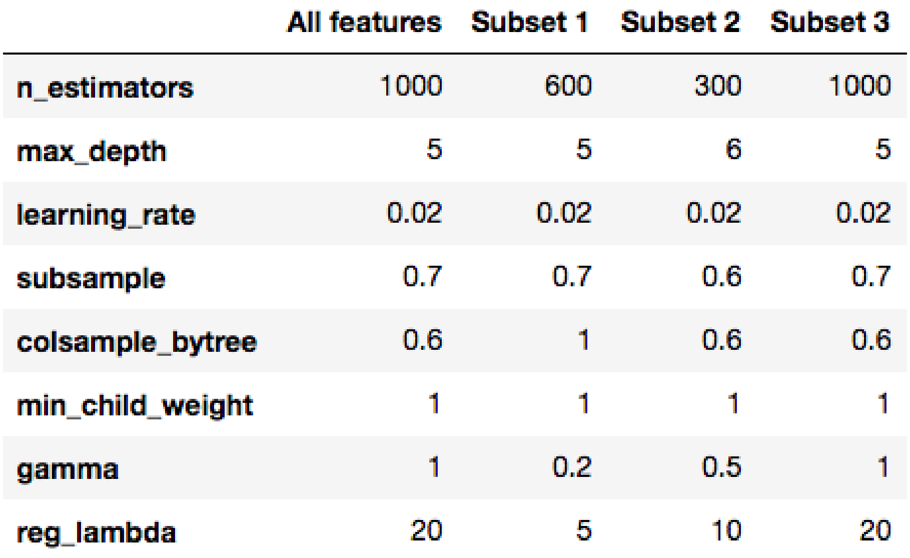


Figure 31: Distribution of differences between residuals for predicted and scheduled durations against actual turnaround times

Table 17 below lists the selected hyperparameters for the four different models. According to the table, models with larger feature sets have resulted in larger number of estimators. Maximum depth, learning rate, subsampling of rows as well as minimum child weight have similar values across models. Instead, colsample_bytree hyperparameter has larger value of 1 for model 2. This could imply that column, or feature, sampling for different trees has not been necessary to avoid overfitting of the model. Regularization parameters gamma and reg_lambda have largest variations in hyperparameter values between models and also as compared to the default values of 0 for gamma and 1 for reg_lambda.

Table 17: Selected hyperparameters for different XGBoost models

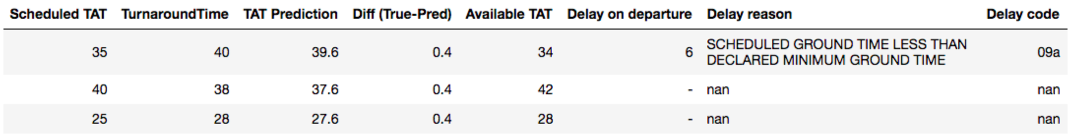


	All features	Subset 1	Subset 2	Subset 3
n_estimators	1000	600	300	1000
max_depth	5	5	6	5
learning_rate	0.02	0.02	0.02	0.02
subsample	0.7	0.7	0.6	0.7
colsample_bytree	0.6	1	0.6	0.6
min_child_weight	1	1	1	1
gamma	1	0.2	0.5	1
reg_lambda	20	5	10	20

Table 18 below displays values of a turnaround where the predictions are close to actual turnaround durations. Difference between true duration and predicted is 0.4 minutes for each row. Based on the columns Available TAT and Scheduled TAT, all the three turnarounds are either close to schedule or before schedule during beginning of the turnaround as the difference between available turnaround time is only few minutes apart from the scheduled turnaround time.

For first row the value of TurnaroundTime is higher than the value of Available TAT which means that the turnaround has lasted longer than the available turnaround time and has therefore departed 6 minutes late. This is also indicated by the delay code which implies that due to certain reason, such as high passenger count, the minimum turnaround duration determined by the manufacturers manuals for this setup is higher than the scheduled turnaround time. Two other rows have been able to depart in time. Actual turnaround durations provided by the column TurnaroundTime are a maximum of five minutes apart from the scheduled turnaround time. This suggests that these three turnarounds might have progressed as expected without surprising occurrences.

Table 18: Turnaround durations for predictions with good accuracy

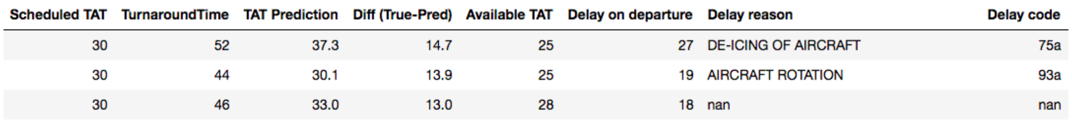


Scheduled TAT	TurnaroundTime	TAT Prediction	Diff (True-Pred)	Available TAT	Delay on departure	Delay reason	Delay code
35	40	39.6	0.4	34	6	SCHEDULED GROUND TIME LESS THAN DECLARED MINIMUM GROUND TIME	09a
40	38	37.6	0.4	42	-	nan	nan
25	28	27.6	0.4	28	-	nan	nan

On the other hand, Table 19 below lists 3 worst predictions where the difference between actual turnaround time and predicted turnaround time have largest positive difference compared to all predictions. Difference between columns Available TAT and Scheduled TAT indicates that all the three turnarounds are late from schedule in the beginning of the turnaround. The actual turnaround times for all the turnarounds are a minimum of 14 minutes apart from the scheduled turnaround times meaning that actual turnaround durations have been longer than expected. Also, actual turnaround times for all rows are larger compared to available turnaround times which indicates late departures.

Difference between actual duration and prediction is over 13 minutes for all the three turnarounds in Table 19. XGBoost model has expected the top row, which is the worst performing turnaround, to last 7.3 minutes more than the scheduled value. According to the column Delay reason, there has occurred a deicing which has been named as the major cause for the delay. For the middle row, machine learning model has expected the turnaround to have similar duration as the Scheduled TAT. Delay reason for the second row is aircraft rotation which indicates that the aircraft has arrived late but besides that, this delay code does not imply any other specified factors related to the delay. Third row does not have a specified delay reason and there is no additional information on why the turnaround time has been 16 minutes more than the scheduled turnaround time. Based on the table it is impossible draw further conclusions about the turnarounds.

Table 19: Turnaround durations for predictions with largest positive difference between actual and predicted values

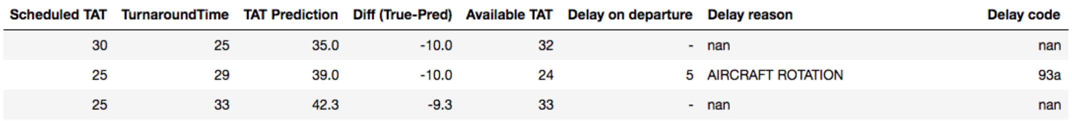


Scheduled TAT	TurnaroundTime	TAT Prediction	Diff (True-Pred)	Available TAT	Delay on departure	Delay reason	Delay code
30	52	37.3	14.7	25	27	DE-ICING OF AIRCRAFT	75a
30	44	30.1	13.9	25	19	AIRCRAFT ROTATION	93a
30	46	33.0	13.0	28	18	nan	nan

On contrary to turnarounds with maximum positive error between true and predicted turnaround times, Table 20 below displays three turnarounds which have the maximum negative difference between said values. Negative difference indicates that turnarounds have in reality been shorter than the machine learning algorithm has expected. First row in the table has a turnaround time 5 minutes less than scheduled turnaround time. In addition, Available TAT is 2 minutes longer than Scheduled TAT. It means that this specific turnaround has arrived before schedule and turnaround process has been accomplished before scheduled departure time. However, the prediction of the first row is 5 minutes apart from the scheduled turnaround time on the opposite direction of the actual turnaround time, resulting in a 10-minute difference between true and predicted values of the turnaround duration. In this case machine learning model has expected the turnaround to last 3 minutes more than the Available TAT, which is 32 minutes.

The two other turnaround rows have longer actual turnaround times compared to the scheduled turnaround times. The middle row has been 1 minute late at the beginning of the turnaround and has departed 5 minutes late due to aircraft rotation. The turnaround of the last row has been started before schedule and it has departed just in time. For both rows the predicted duration is to the same direction as the actual duration, anticipating a longer than scheduled duration, but both rows overestimate the actual turnaround duration at least 9 minutes.

Table 20: Turnaround durations for predictions with largest negative difference between actual and predicted values



Scheduled TAT	TurnaroundTime	TAT Prediction	Diff (True-Pred)	Available TAT	Delay on departure	Delay reason	Delay code
30	25	35.0	-10.0	32	-	nan	nan
25	29	39.0	-10.0	24	5	AIRCRAFT ROTATION	93a
25	33	42.3	-9.3	33	-	nan	nan

Table 19 and Table 20 with maximum differences between actual and predicted turnaround durations provide some information on the turnarounds but regardless of that, the reason for large errors is not evident. More details will be discussed related to these turnarounds in the section 5.6.2. As mentioned in the methodology chapter, part of the delay codes is included in the dataset after the preprocessing steps. This is due to the fact that removal of rows with certain delay codes, such as aircraft deicing, could distort the effect of deicing in general especially as it is included as an input feature. For some delay codes such as aircraft rotation it is not certain if the turnaround duration or its subprocesses are somehow affected by the indirect reasons such as the mere fact that the aircraft is late.

To distinguish the effect of the delay codes on the distribution of the residuals by the XGBoost model 1, Figure 32 below illustrates residuals for turnarounds with and without delay codes. Figure suggests that the accuracy of XGBoost for turnarounds with delay codes is weaker compared to turnarounds without delay codes. For predictions of turnarounds with delay codes, the distribution contains residuals which are more widely dispersed. Especially for residual values between range 2.5 minutes and 12.5 minutes, the distribution with delay codes has a larger density which means that turnarounds with delay codes are more likely to have a positive residual with a larger magnitude compared to turnarounds without delay codes. This leads to a conclusion that there exist phenomena behind turnarounds with the remaining delay codes that the machine learning model is not able to explain.

This occurs despite the idea of leaving selected delay codes in the dataset for the reason that removal of such rows could also remove essential relationships between variables. Next section reviews the visualizations of the dataset based on SHAP values and at the end of next section more interpretability for turnarounds in Table 18, Table 19 and Table 20 will be provided.

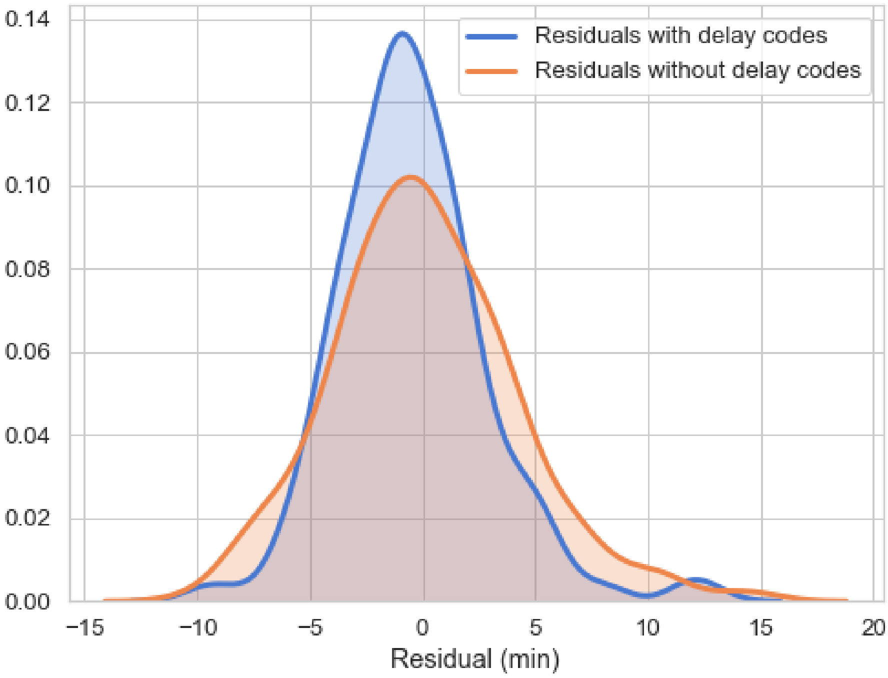


Figure 32: Residuals between actual and predicted durations for turnarounds with and without determined delay codes

5.6 Interpretation of predictions with SHAP

SHAP is the main method in this thesis to supply interpretability for the XGBoost models. This section starts with the review of global effects of different features in the dataset followed by the visualization of interactions and dependencies between features. Finally, the interpretability for individual turnarounds, which were discussed in the previous section, will be reviewed.

5.6.1 Global feature effects

Figure 33 below visualizes SHAP feature importance values for individual features as bars and cumulative feature importance as a curve. According to Molnar (2019), SHAP feature importance values present the global importance for features by averaging Shapley values for for each feature across the dataset. Based on the feature importances in Figure 33, two most important features are PaxOnBoardOut and Deice. Third most important feature is Available TAT and from the fourth important feature PaxOnBoardIn onwards the trend of the feature importance value decreases steadily. Features FuelNeed, MailIn and MailOut have nearly non-existent importance values. One note worth pointing out from the figure is that all the features representing aircraft variants have low importance values. This indicates that difference provided by different aircraft variants in the dataset is either not relevant or the properties of different aircraft variants are provided by other features to adequate extent.

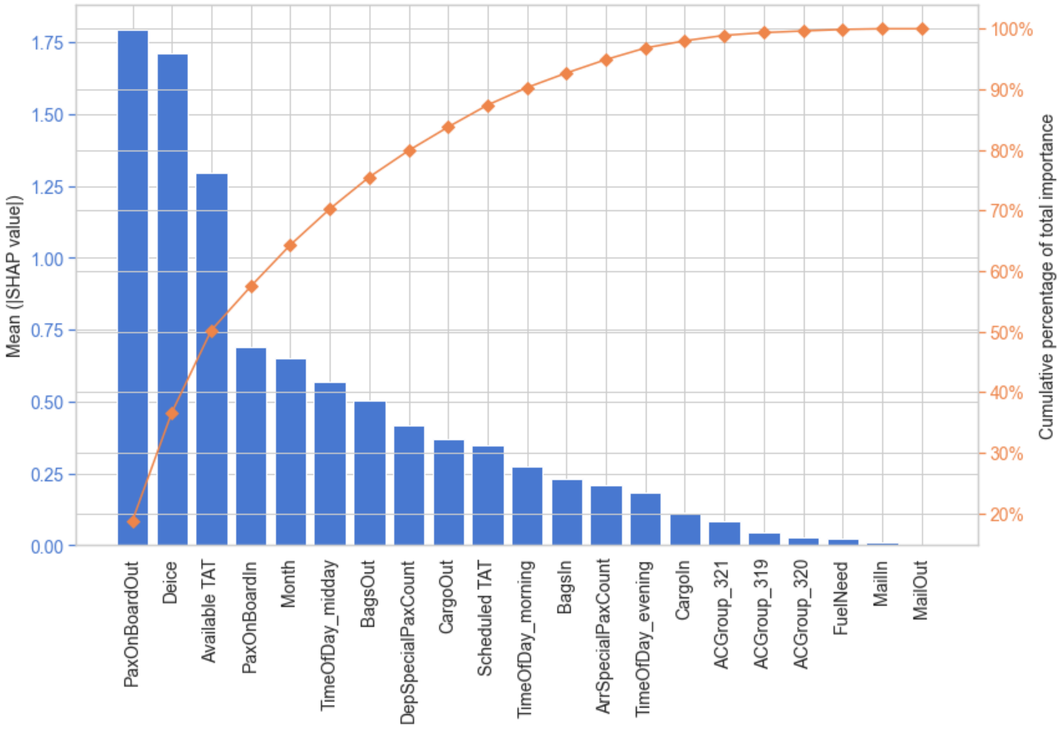


Figure 33: SHAP feature importance values

Based on the application of the Pareto rule, which was discussed in the previous chapter, the subset of features that corresponds to the 80 percent of the total cumulative importance consists of PaxOnBoardOut, Deice, Available TAT, PaxOnBoardIn, Month, TimeOfDay_midday, BagsOut and DepSpecialPaxOut as seen from Figure 33. All of these features were also present at least in one or the other of the feature subsets 1 and 2 selected by the RFE method.

SHAP summary plot visualizes information related to feature importances and feature effects (Molnar, 2019). In the Figure 34 below, y-axis consists of features ordered by feature importance values from highest to lowest. Effect of a single feature is shown on the x-axis which includes all the observations from the dataset. Color coding represents the value of a single observation for that feature so that red refers to a high value and blue refers to a low feature value.

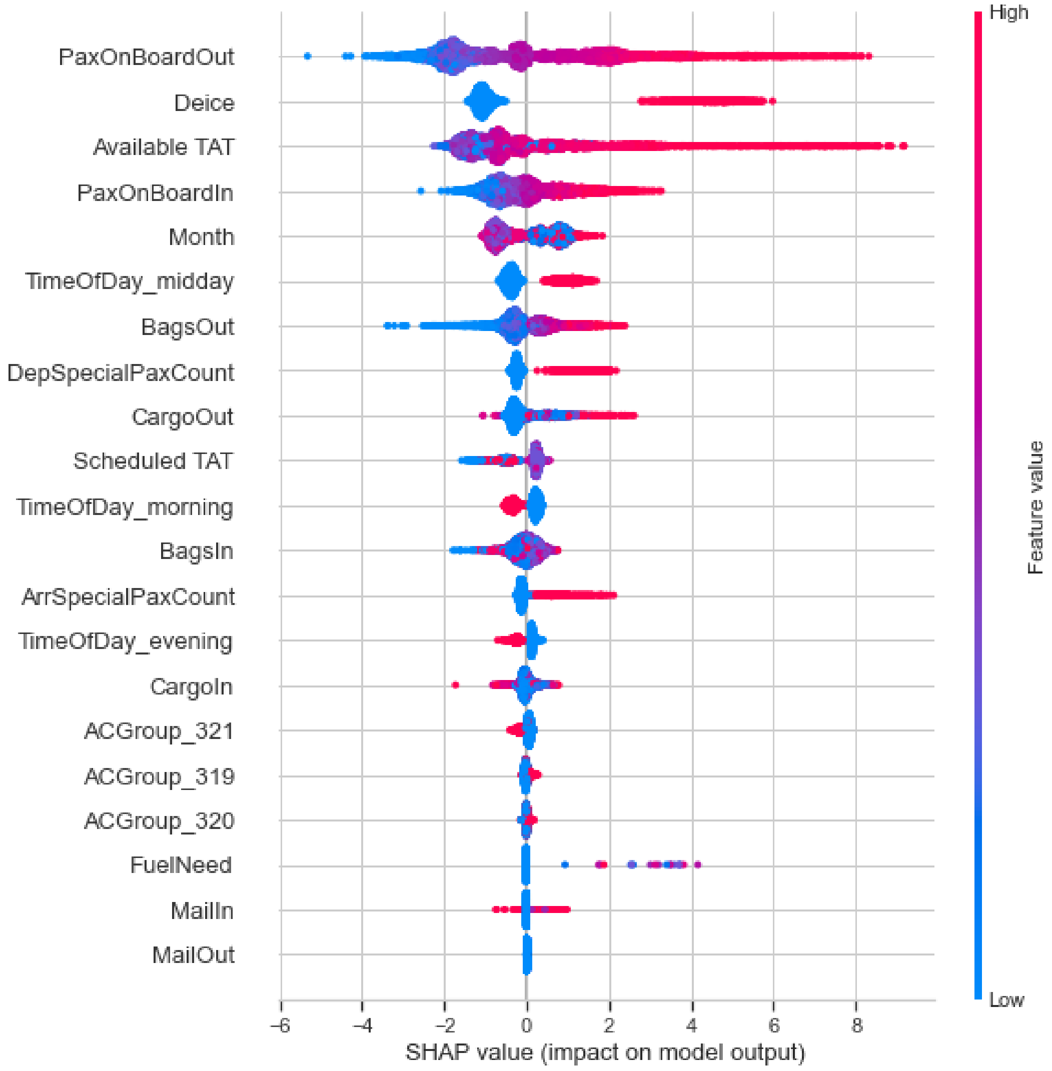


Figure 34: SHAP summary plot

PaxOnBoardOut is the most important feature in the prediction of a turnaround and based on the figure, the scale for PaxOnBoardOut SHAP values is wide. Observations with medium to high values for the feature PaxOnBoardOut can have an effect on the model output ranging from -1 minutes to even 8 minutes whereas observations with low PaxOnBoardOut values are concentrated with a shorter range from approximately -4 minutes to -2 minutes. This can be interpreted so that even though PaxOnBoardOut is a major determinant in the turnaround time prediction, the effect of a certain departing passenger number on the prediction output is dependant on the other features as well.

Deice feature has a greater concentration of observations on certain areas according to the figure. If deicing is not performed, the figure indicates that average effect on the output is -1 minute whereas the performance of deicing has a ranging effect from approximately 3 minutes close to 6 minutes. It is good to note that the average effect is negative instead of zero if the deicing is not performed.

Observations for feature Available TAT also disperse on a wide range. However, in Figure 34 observations with low values for Available TAT are concentrated on a small area whereas high values are widely dispersed. Feature PaxOnBoardIn has a varying effect from approximately -3 minutes to 3 minutes and similarly to PaxOnBoardOut, observations with low values for arriving passengers are more densely packed compared to high values. SHAP values for feature Month are hard to interpret from the figure due to the numerical representation of monthly values where, in reality low values such as 1 is most probably more closely related to high value of 12 regarding its effect on the model output.

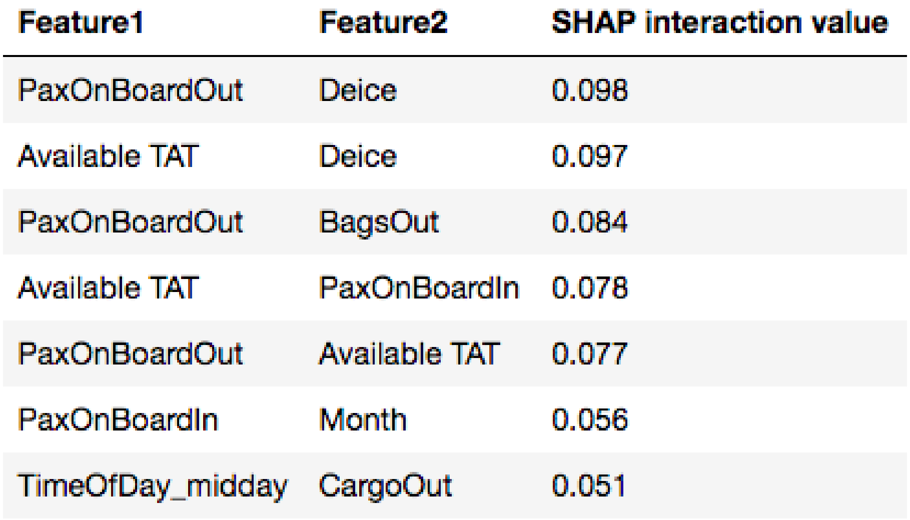
Similarly to Deice, the feature TimeOfDay_midday has a boolean value and its effect on the model output varies from almost -1 minute to 1.5 minutes. The effect of BagsOut feature resembles the effect of feature PaxOnBoardOut which is justified by their close connection, as discussed previously. Features DepSpecialPaxCount and ArrSpecialPaxCount both have similar effects on the model output by providing a small negative contribution with observations where special passengers are not on board and a wider range of positive contributions up to over 2 minutes when one or more special passengers are either arriving or departing from airport A.

Feature CargoOut has a small negative contribution of approximately -1 minute and maximum positive contribution of 2.5 minutes. For this feature it is worth to note that between the range of 0 minutes and 1 minute observations with high and low values for CargoOut seem to be concentrated and overlapping. This could indicate that with some other feature combination the positive effect of low CargoOut values is higher than the effect of high CargoOut values even though on average low CargoOut values are centered on the other side of the zero effect axis. Features Scheduled TAT and BagsIn have overlapping values and further conclusions for these features based on the SHAP summary figure are hard to be drawn. TimeOfDay_morning and TimeOfDay_evening features show clear and concise effect on the model output but magnitude of the effect of these features such as the effect of all the remaining features is less than one minute.

Most of the remaining features show an unambiguous and reasonable effect on the model output excluding the features CargoIn, FuelNeed and MailIn. CargoIn has a negative impact on the output for high CargoIn values. This could be due to the reason that arriving cargo either replaces bags and is on average a faster operation compared to unloading of bags or that arriving cargo happens in connection with another factor and the end result provides a negative effect on the output. FuelNeed, on the other hand, often has a contribution of 0 minutes on the output of the model. However, based on the SHAP summary, when fuelling is performed, it has a positive contribution of several minutes. This is logical as it could be that anytime when fuelling is performed, it is an additional subprocess. Indication for MailIn demonstrates that when MailIn feature has value of 0, its contribution is also 0 minutes but for higher values the contribution can be both positive and negative.

Next, SHAP interaction values will be presented followed by the visualization of dependencies for single features with potential interaction values. Table 21 contains highest interaction values in the dataset above the threshold 0.05 minutes. Interactions with strength less than 0.05 minutes were not included as they were considered weak. Similarly to the interactions provided by the H-statistic, feature pairs that visually demonstrated interaction were selected to be displayed from Table 21. Such feature pairs are PaxOnBoardOut and Deice, Available TAT and Deice, Available TAT and PaxOnBoardOut and CargoOut and TimeOfDay_midday. In addition, feature pairs PaxOnBoardOut and TimeOfDay_morning, PaxOnBoardOut and TimeOfDay_midday as well as individual features BagsIn and CargoIn will be visualized.

Table 21: SHAP interaction values for feature pairs



Feature1	Feature2	SHAP interaction value
PaxOnBoardOut	Deice	0.098
Available TAT	Deice	0.097
PaxOnBoardOut	BagsOut	0.084
Available TAT	PaxOnBoardIn	0.078
PaxOnBoardOut	Available TAT	0.077
PaxOnBoardIn	Month	0.056
TimeOfDay_midday	CargoOut	0.051

SHAP dependence plots are used for global interpretation of individual features where SHAP values for individual observations are on the y-axis and feature values are presented on the x-axis (Molnar, 2019). In addition, another feature can be added on the plot to reveal interactions and a scale on the right side of the plot displays respective values for the interacting feature. Even though dependence plot consists of individual observations, it is a global interpretation tool as it reveals global trends from the data. In the dependence plots below, in some occasions the exact values have been replaced with indicator “high” and “low” to indicate the magnitude of values on the axis. These are part of the means to prevent visualization of exact values which are considered confidential.

Figure 35 below presents a dependence plot for the feature PaxOnBoardOut supplemented with interaction effects with the feature Deice. According to the plot, the relationship between feature PaxOnBoardOut values and the corresponding SHAP values is nearly linear and SHAP values have a range from approximately -5 minutes to over 8 minutes. Interesting observation is that the effect of deicing on the SHAP values varies based on the departing number of passengers. For passenger count on the left side of the x-axis towards the low values, the observations with deicing have an addition in the SHAP values whereas observations towards higher passenger counts with deicing have lower SHAP values compared to high passenger counts without deicing.

One explanation for this observation could be that if deicing can be performed simultaneously with boarding, for short boarding times the deicing process can take longer than the boarding process and in such cases turnarounds with deicing have on average a longer turnaround duration. However, for turnarounds with both large number of passengers and deicing the lesser contribution through the SHAP values compared to turnarounds without deicing is not evident. This could be related to the calculation of SHAP values which may in some occasions result in a situation that a feature with no real effect may instead have a small negative effect, as discussed in section 4.5.1 earlierly. When passenger count is close to the highest values, aircraft variant is always A321 due to its passenger capacity which means that the difference is not based on different aircraft variants.

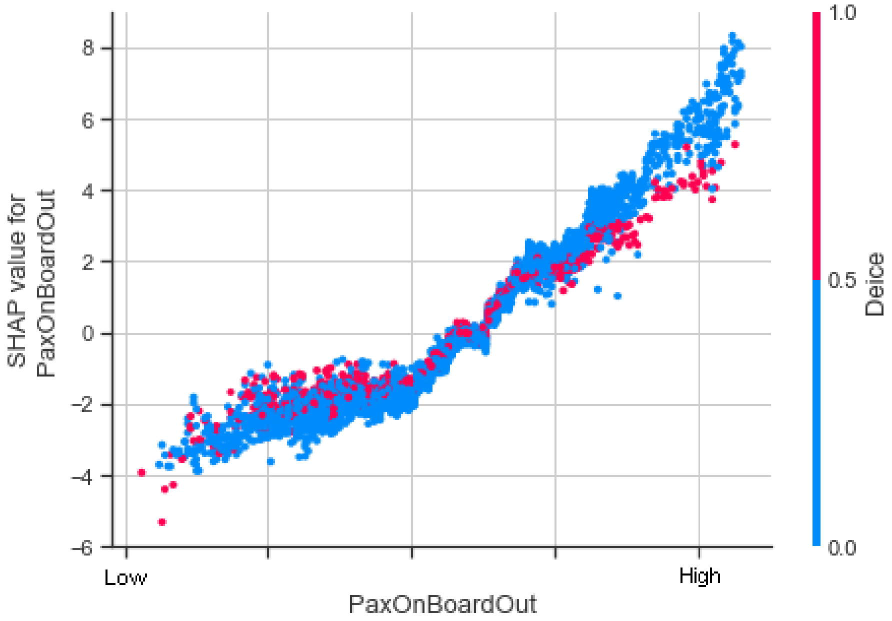


Figure 35: SHAP dependence plot for feature PaxOnBoardOut with Deice interactions

Figure 36 below displays a dependence plot for PaxOnBoardOut but this time including interaction effects with the feature TimeOfDay_morning. Figure indicates that similarly to deicing, turnarounds with a small amount of passengers have a larger SHAP value when the turnaround is performed during morning. However, this trend changes from the midway of the x-axis so that turnarounds performed in the mornings with higher passenger values have a decrease in the SHAP values. For turnarounds with highest passenger counts (approximately last fifth of the x-axis) there is no similar distinct difference. Figure 5 in the introduction of different variables indicated that most of the morning turnarounds are performed with A320 which explains the sparsity of morning turnarounds with highest passenger counts.

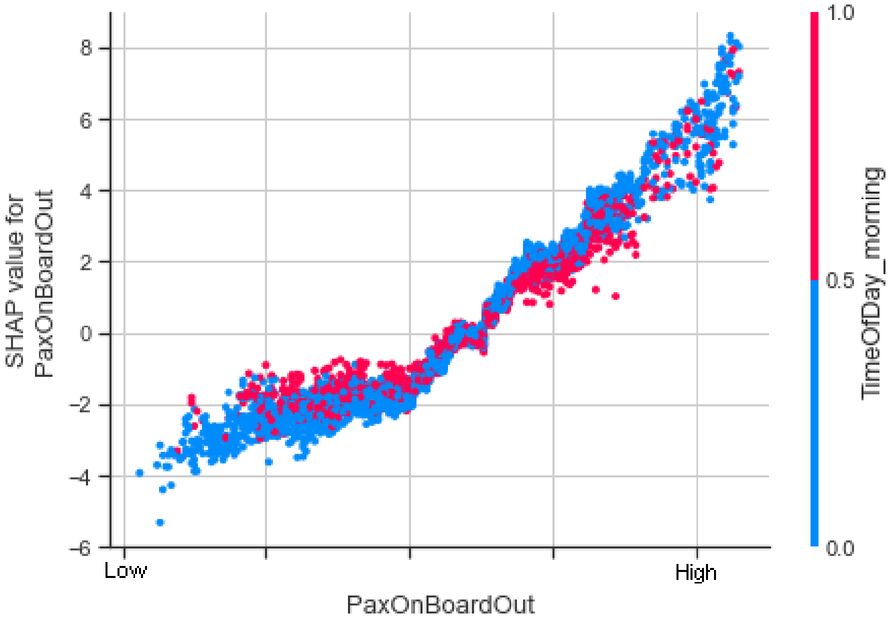


Figure 36: SHAP dependence plot for feature PaxOnBoardOut with TimeOfDay_morning interactions

Interaction effects for the feature PaxOnBoardOut with the feature TimeOfDay_midday are shown in Figure 37 below. During middays the turnaround is almost equally likely performed with all the different aircraft variants according to Figure 5 in section 4.6.1. Figure below indicates that for turnarounds performed during middays with low passenger counts the contribution through SHAP value is less than for turnarounds performed during other times of a day with similar passenger counts. It is worth to note that for the lowest passenger counts there are no turnarounds during middays. For turnarounds with higher passenger counts the TimeOfDay_midday feature creates an increase in turnaround duration but with the most highest departing passenger counts there is no similar recognizable trend. The difference created by the feature TimeOfDay_midday, which is a maximum of few minutes, could indicate that during middays a procedure or some other factor is performed differently compared to other times of day and this change leads to the difference in turnaround times.

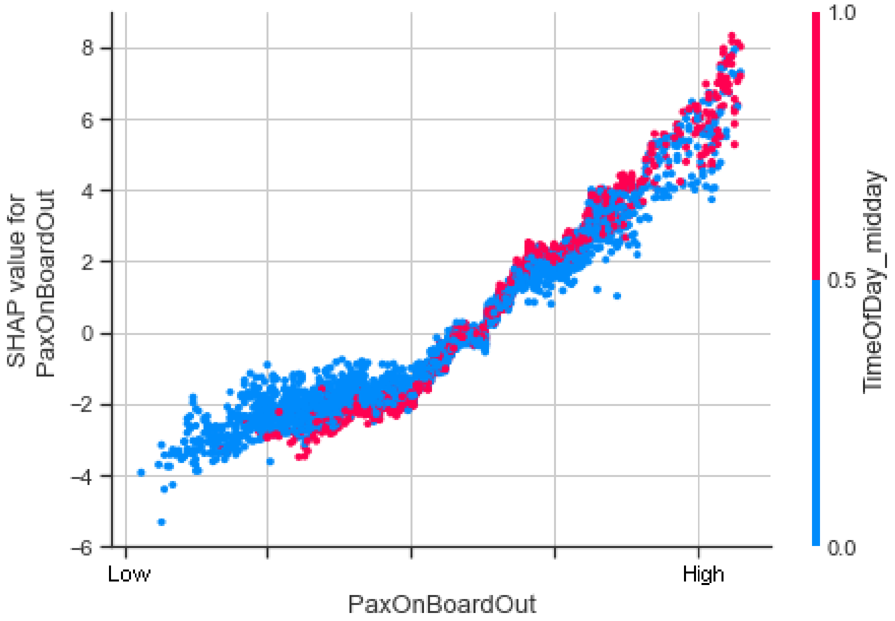


Figure 37: SHAP dependence plot for feature PaxOnBoardOut with TimeOfDay_midday interactions

Figure 38 below shows a dependence plot for the feature Available TAT with interactions with the feature Deice. The plot provides insight why the feature Available TAT has been considered so significant in the determination of feature importance values. The range of Availble TAT value from approximately 35 minutes up to 60 minutes corresponds a contribution difference of over 8 minutes in the SHAP values. On the other hand, values of Available TAT less than 35 minutes correspond a range of SHAP values from approximately -2 minutes to 2 minutes. The mean values for the target variable TurnaroundTime varies from 33 minutes to 34.5 minutes depending on the aircraft variant and most typical Scheduled TAT value is 30 minutes for all aircraft variants.

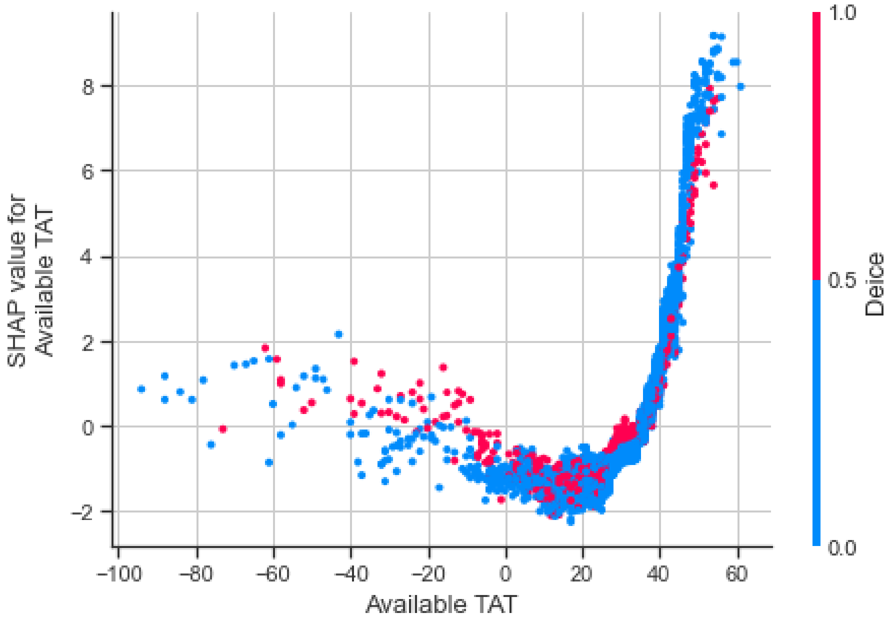


Figure 38: SHAP dependence plot for feature Available TAT with Deice interactions

In practice the large effect on the SHAP values provided by Available TAT could have a following explanation: Available TAT values below 35 minutes correspond a situation when an aircraft is just in time or arrives late which means that there is no spare time in the schedule and the aircraft will leave as soon as possible. On the other hand, when Available TAT is over 35 minutes, it is likely that most of the turnaround time is used despite the slack time as due to the released schedule the aircraft cannot depart many minutes ahead of schedule. Large Available TAT values therefore increase the duration of the turnaround and have a large effect on the SHAP values.

The concentration of observations in the Figure 38 for Available TAT between 30 and 35 minutes indicates that when deicing is performed, SHAP value increases for a turnaround. Same trend can be identified for Available TAT values less than 0 minutes. For large values of Available TAT, observations with deicing have a smaller SHAP value compared to observations without deicing.

Below in Figure 39, interaction effects for Availble TAT with PaxOnBoardOut are visualized. According to the figure there are no remarkable interactions between these two features excluding the part around the Available TAT value of 20 minutes. In that part of the plot, turnarounds with large number of departing passengers have a larger SHAP value compared to observations with fewer number of passengers with a maximum difference of less than 2 minutes. Interesting point is, however, that for low Available TAT values which correspond to turnarounds that are late at arrival, there is no clear trend for the difference between low and large number of passengers.

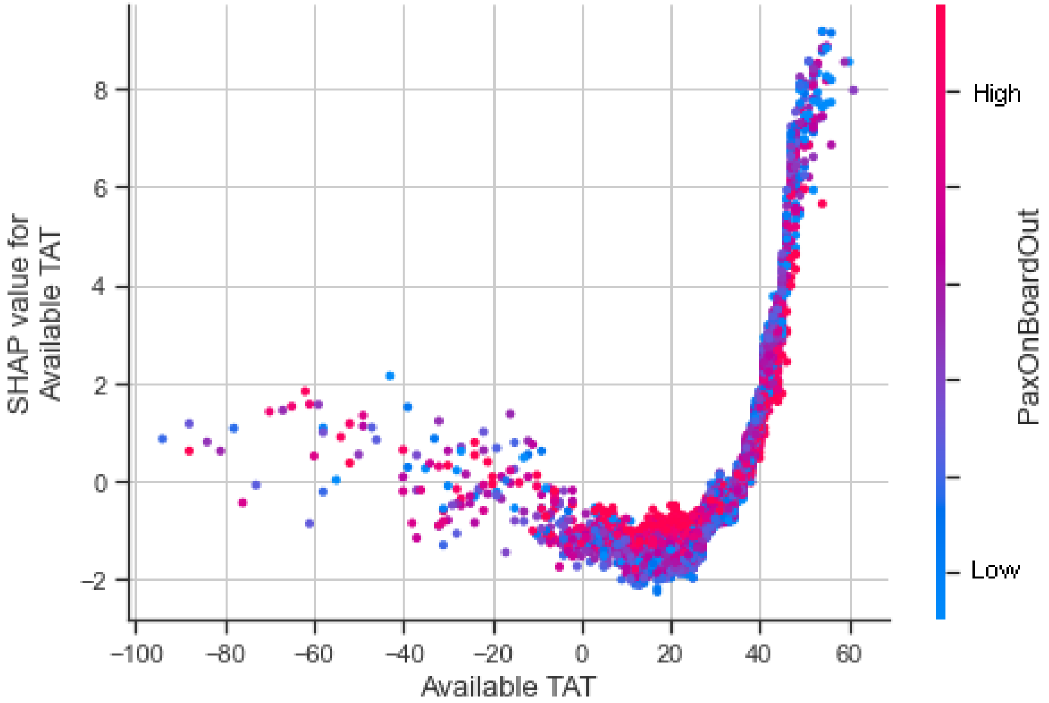


Figure 39: SHAP dependence plot for feature Available TAT with PaxOnBoardOut interactions

Dependency plot and interactions for feature CargoOut with the feature TimeOfDay_midday are shown below in Figure 40. Figure demonstrates that the effect of CargoOut, based on the SHAP values, has a range from approximately -1 minutes to over 2.5 minutes with widely dispersed observations on the plot and no clear trend. However, a clear trend can be seen between observations which have occurred during midday compared to other times of day. Based on the plot, SHAP values for turnarounds with outgoing cargo and which have occurred during midday are mostly lower than for turnarounds which have occurred during other times of day. Maximum difference for SHAP values is 3 minutes which can be seen for CargoOut values around approximately halfway of the x-axis. This supports the idea that an additional unknown factor happening during middays makes a distinct difference in turnaround times and in this case it happens for almost all of the flights with outbound cargo.

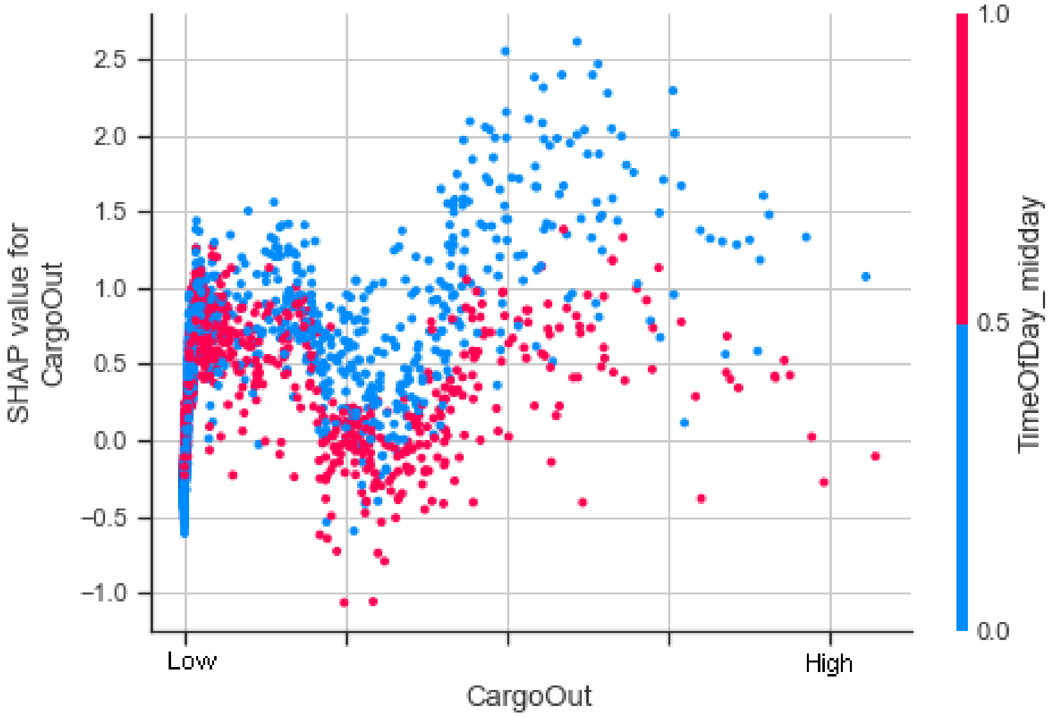
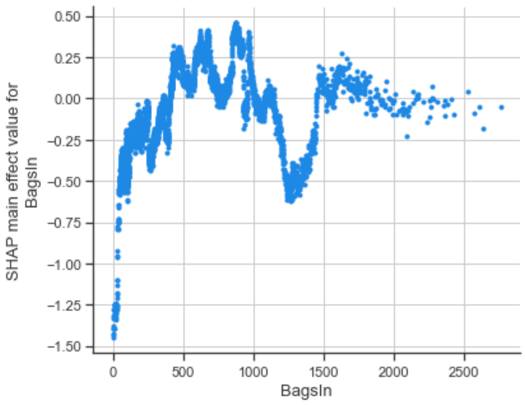


Figure 40: SHAP dependence plot for feature CargoOut with TimeOfDay_midday interactions

Below two different dependence plots, Figure 41 and Figure 42, with features BagsIn and CargoIn provide only the main effects of these features, in other words, interaction effects with other variables are not included. From both figures a clear single consistent trend is difficult to be identified but both plots contain periodical up and downward trends. The maximum difference of the SHAP values between a bottom and a top are less than a minute in both figures but nevertheless, they give potential indication for an underlying mechanism which could be related to cargo unloading during arrival.



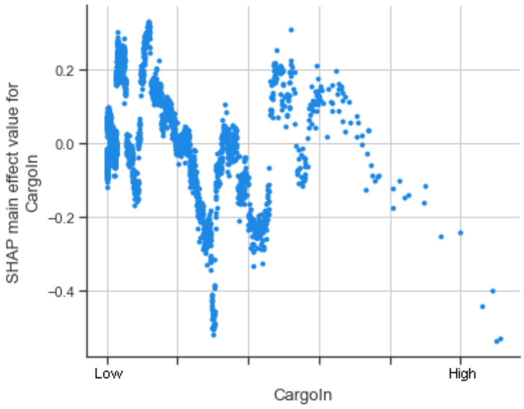


Figure 41: Main effects of the feature BagsIn

Figure 42: Main effect of the feature CargoIn

5.6.2 Local feature effects

Local feature effects refer to the effects of different features on individual predictions. In this section local feature effects will be visualized with waterfall plots provided by the SHAP framework. The individual predictions discussed in this section are the same as in section 5.5 above. Waterfall plot is based on the idea that an expected model output is derived from the background dataset and contributions of all the features are shown on their own rows (Lundberg, 2018). By summing the expected output and contributions, the predicted output is shown on top of the waterfall figure. Left side of the figure indicates the values of each feature.

Figure 43 below visualizes the prediction for a turnaround discussed in section 5.5 where the accuracy was considered to be good as the actual turnaround time was 40 minutes and predicted turnaround was 39.6 minutes. Scheduled turnaround duration was 35 minutes. From the figure it can be concluded that feature PaxOnBoardOut with a value of 210 passengers is the main driver of the predicted time with a value of +7.51 minutes (note: the passenger number is shown to be 210 even though the maximum capacity is 209, which could be due to passengers that are seated on a jump seat/a seat for cabin personnel). Second most important driver is the BagsOut feature with a contribution of +1.8 minutes. The contributions of the remaining features are all less than a minute. However, the waterfall plot succesfully visualizes how the cumulative effect of multiple small contributions, which describe different details of the turnaround process, results in a total effect of multilple minutes.

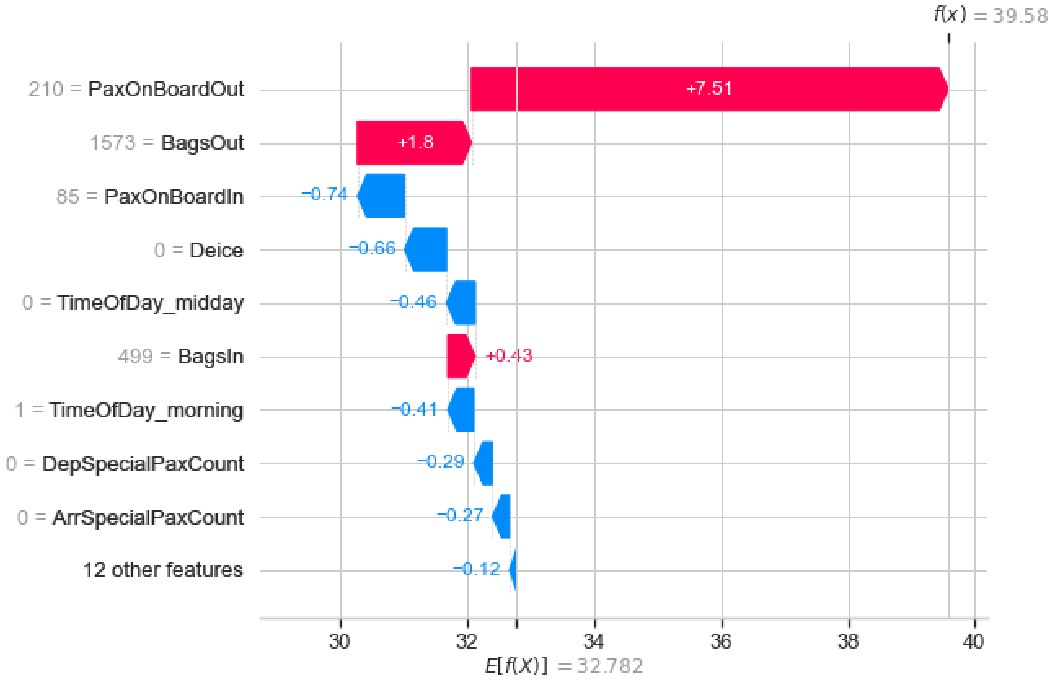


Figure 43: Waterfall plot for a prediction with good accuracy

Waterfall plot visualized in Figure 44 below, on the other hand, illustrates the turnaround with the largest positive difference between the actual turnaround time (52 minutes) and the predicted turnaround time (37.3 minutes). Delay for this turnaround is due to deicing of aircraft based on the table in section 5.5 and according to Figure 44, deicing has a contribution of +4.46 minutes on the predicted turnaround duration. Scheduled turnaround time for this particular turnaround was 30 minutes. In addition to Deice, there does not seem to be any other features either with a large contribution on the prediction or features with extreme feature values.

Factors related to Deice and the difference between actual and predicted durations imply that the prediction is on the right direction but insufficient in taking into account the duration of the deicing process. Dataset reveals that for this turnaround the deicing procedure has been a two-step procedure consisting of both deicing and anti-icing. However, deicing steps are not specified for the XGBoost model at the moment and even so, the maximum effect of deicing based on global SHAP values would not correspond to the actual length of the deicing in this case. A turnaround as such would need more data to be provided so that different factors could be included in the predictions.

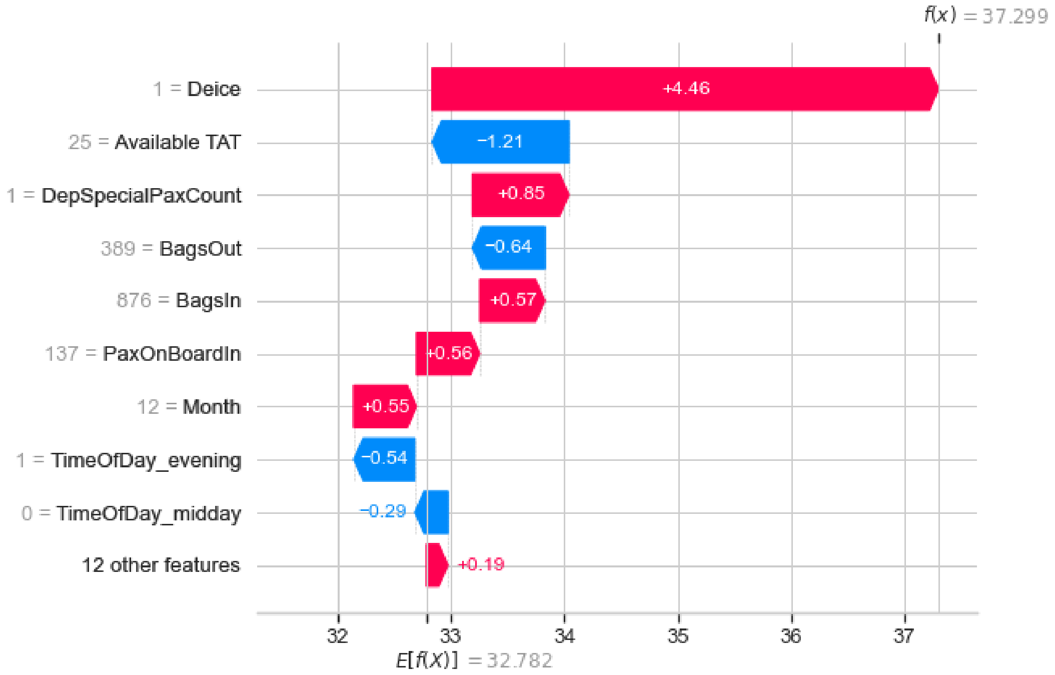


Figure 44: Waterfall plot for a prediction with largest positive difference with actual turnaround time

Figure 45 below, describes a turnaround with the second largest positive difference between actual turnaround and predicted turnaround time. In this case the predicted duration (30.1 minutes) has been close to the scheduled turnaround duration which was 30 minutes. Actual turnaround time has been 44 minutes which means there has been almost 14-minute difference between the actual value and the prediction. Figure below does not indicate any extreme values or reasons for the high deviation from the scheduled turnaround time and from the machine learning model's point of view the turnaround has been uneventful. Based on the table in section 5.5, the delay code for this turnaround has been aircraft rotation which means that the aircraft has arrived late resulting in an available turnaround time of 25 minutes. As a delay code, aircraft rotation (93) does not convey information about the possibly effect it has on the turnaround procedures.

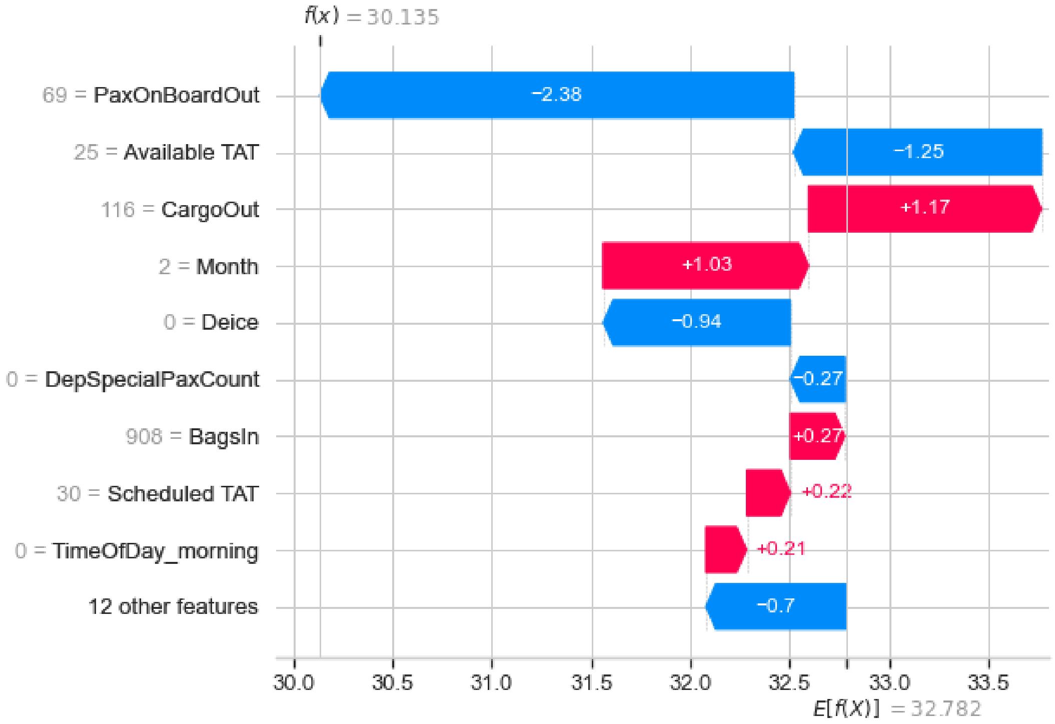


Figure 45: Waterfall plot for a prediction with second largest positive difference with actual turnaround time

Last two figures, Figure 46 and Figure 47 of local feature effects demonstrate the predictions for turnarounds with two of the largest negative differences between actual and predicted turnaround times. Figure 46 below has the largest negative difference with actual turnaround time of 25 minutes and predicted turnaround time of 35 minutes. Scheduled turnaround time for this turnaround was 30 minutes which means that the prediction has been made to the wrong direction as exceptionally the actual turnaround time has been shorter as the scheduled time. Available TAT for this turnaround has been 32 minutes which indicates that the aircraft has arrived 2 minutes before scheduled time and the machine learning model has considered this with contribution by the Available TAT of -0.46 minutes.

Largest contribution for this turnaround comes from feature CargoOut (+2.12 minutes). Feature PaxOnBoardOut, which has been a major contributor in many cases, has a value of 126 passengers, which is not indicated in the figure, but can be considered as a more or less neutral value for that feature. Second and third highest contributions come from features Deice and Month with an effect where they cancel each other. For this turnaround there are no delay codes as the aircraft has been able to depart before schedule. For the machine learning model, a turnaround such as this, could have appeared as a turnaround with minor reasons for a small delay. Seems that the model has not been able to anticipate a short turnaround time which could also be due to the interaction of multiple features which are not evident.

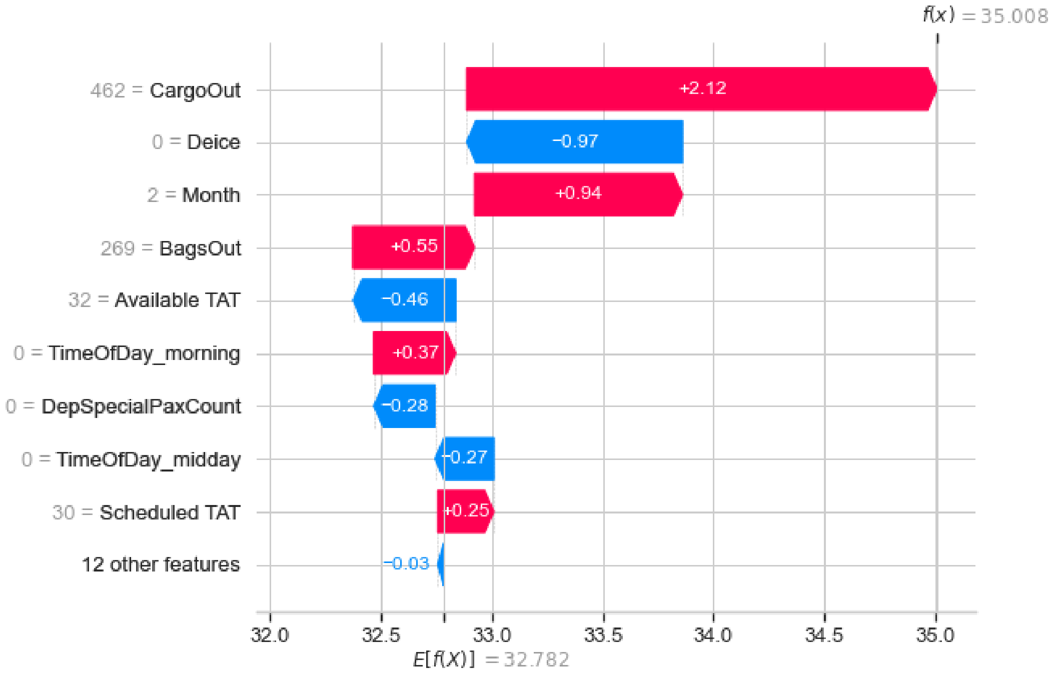


Figure 46: Waterfall plot for a prediction with largest negative difference with actual turnaround time

Figure 47 below illustrates the turnaround with second largest negative difference between actual and predicted turnaround times. The actual turnaround time has been 29 minutes whereas the predicted turnaround has been 39 minutes and scheduled turnaround time 25 minutes. In this case the prediction has been made to the right direction with insufficient magnitude. Available TAT has a value of 24 minutes which means that the aircraft has arrived late. Aircraft rotation has been given as a delay reason for this turnaround.

The machine learning model has given a large emphasis for the feature Deice with a contribution of +5.06 minutes. As seen from a previous example, contribution for deicing approximately 4 or 5 minutes is a typical value. Second largest contribution comes from the number of departing passengers which is 142. Based on these factors, it may not be feasible to say that the model has overreacted in this prediction, as for example the effect of Deice is within reasonable limits, but instead the model has either underreacted by not giving enough contribution for features that would decrease the predicted time or does not have the necessary data to be able to create sufficient subtractions for the base value.

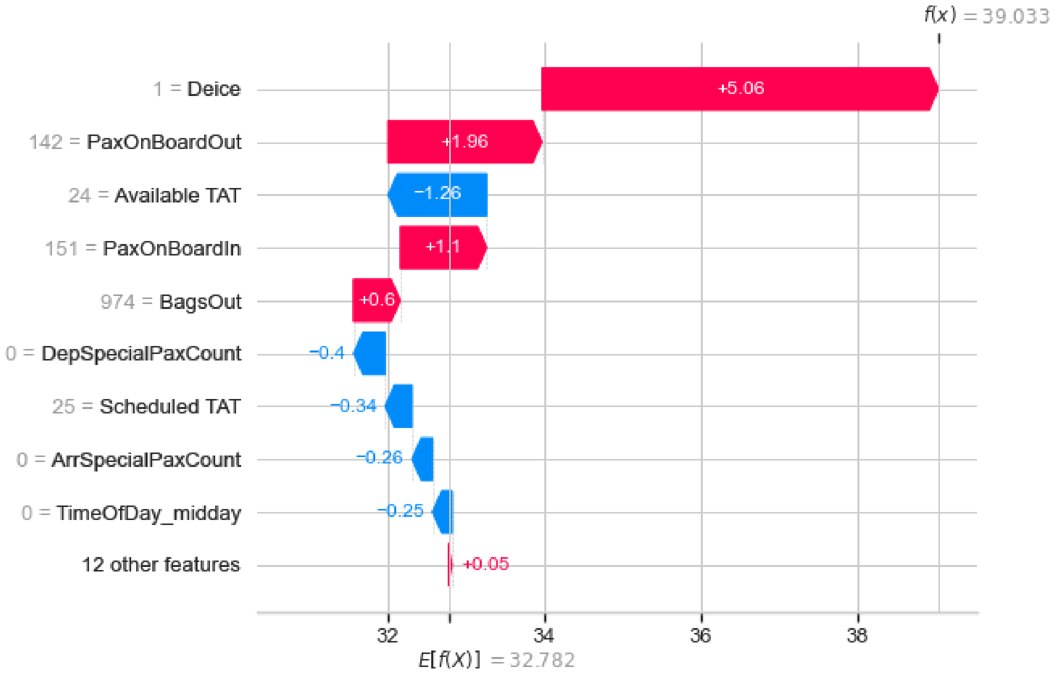


Figure 47: Waterfall plot for a prediction with second largest positive difference with actual turnaround time

As a summary for the local interpretations based on these five examples and Table 18, Table 19 as well as the Table 20 discussed before in the section 5.5, the machine learning model is good in the prediction of so-called standard cases. Example turnarounds where the predictive ability of the machine learning model was good, values for scheduled and actual turnarounds durations are a maximum of 5 minutes apart from each other. On the other hand, examples with the largest positive differences between actual and predicted durations had a minimum of 14 minutes between scheduled and actual duration meaning that the actual turnaround duration was in all cases considerably longer than usual. This indicates that the XGBoost model could not anticipate the magnitude of such turnaround lengths and there were no extreme feature values present which could have explained the long actual duration. The example turnaround with the largest positive difference and a delay code “de-icing of aircraft” had a difference of 14.7 minutes between the actual and predicted duration. For this turnaround the machine learning model had given an additional +4.46 minutes contribution on the prediction due to deicing which means that the contribution was not enough and that there may be other issues behind that specific delay.

For the turnarounds with largest negative differences, the actual turnaround values were a maximum of 8 minutes apart from the scheduled durations. For some reason in all the cases, with and without delays and delay codes, XGBoost model overestimated the durations. No extreme feature values were present in these turnarounds excluding the feature CargoOut which brought an increase of +2.12 minutes on one of the turnarounds and deicing which added +5.06 minutes on the other. However, for those two turnarounds the difference between actual and predicted durations is 10 minutes which is only partly explained by these features.

Based on the examples, for large positive differences the machine learning model is not capable of giving enough contribution to reach a duration as high as the actual and for large negative differences the machine learning model overestimates the durations. In general, all the five examples demonstrated that there are typically only a few features that act as the main drives for a prediction such as PaxOnBoardOut or Deice. In most of the cases the remaining features contributed only through minor adjustments on the predictions and their cumulative contribution was a maximum of few minutes. This can be interpreted so that in many cases the few main drivers are capable of explaining the turnaround prediction to a satisfying degree but in the extreme cases the few main drives are either not capable of giving enough weight through these few features on the predictions or they give too much weight. Especially in examples with large negative differences there were no other features that could have compensated the overestimation caused by these few significant features.

One potential conclusion, which can be drawn based on this discussion, is that more features are needed to explain especially the extreme cases when the turnaround time is exceptionally long or when, based on only few features, the model overestimates the turnaround duration. At this point it is unclear what are all the direct or indirect consequences of a late arrival. This is an example of a phenomenon which could be provided for the model in more detail, in addition to the feature Available TAT, which can now reveal the late start of a turnaround. However, quantifying such consequences might be hard, if they exist. Next, the summary of results will be provided followed by the conclusions of this thesis.

5.7 Summary of results

Results chapter has provided a walkthrough and discussion related to correlations and multicollinearity between variables, exploratory data analysis through bivariate visualizations, calculation of H-statistic to inspect interactions, feature selection, assessment of the performance of trained models and finally the interpretability analysis with SHAP. This section contains a summary of the relevant results based on the walkthrough.

Main findings from the correlation part included the moderate correlations between variables TurnaroundTime and PaxOnBoardOut, TurnaroundTime and BagsOut as well as the weak correlation between TurnaroundTime and

Deice. Arriving flow of people and bags have only a weak correlation with TurnaroundTime and they also have a weak negative correlation with the feature CargoIn. Target variable TurnaroundTime has a weak correlation with TimeOfDay_midday, even weaker correlation with TimeOfDay_morning and negligible correlation with TimeOfDay_evening. Correlation table in section 5.1 indicates that based on moderate correlations, cargo arrives to airport A during mornings when, on the other hand, passengers and bags are less likely to arrive. Instead, passengers and bags are more likely to arrive during evenings. Weak correlation indicates that passengers and bags are likely to depart during middays but not during the evenings. Also, cargo is most likely to depart from airport A during middays, but not likely to depart during mornings likewise to departing passengers and bags.

Bivariate visualizations indicate that during winter season turnarounds have, on average, the longest durations and that effect is only partly based on the deicing subprocess. On average deicing prolongs a turnaround process and the difference between actual turnaround time and scheduled turnaround time is more likely to be positive. However, the effect of deicing on turnaround time can be seen more clearly on short turnarounds.

Visualizations also confirm that turnaround time is affected more by the departing bags and passengers compared to arriving bags and passengers. On average largest number of departing passengers is during middays which is also when on average largest amount of cargo departs. Midday is the time of day when turnarounds on average have the longest durations. Number of arriving passengers is largest during the evenings.

Interactions based on the H-statistic demonstrate three different features that interact with the variable Season_Winter. These features are CargoOut, BagsOut and PaxOnBoardIn. In addition, interaction exists between the amount of departing bags and number of departing special passengers.

Feature selection provided by RFE with two different models used in the selection process resulted in two subsets of features. Features that were present in both subsets were PaxOnBoardOut, BagsOut, Deice, Month, TimeOfDay_midday and Available TAT and the optimal size of a subset was 9 features. Results from the RFE method were benchmarked against the subset given by the application of the Pareto rule on the feature importance values provided by SHAP. This comparison demonstrated that subsets provided by RFE and SHAP are similar and they also share the most important features.

Fully trained XGBoost models, excluding the model without features Available TAT and Scheduled TAT, reached a range of explained variance from 0.56 to 0.60. Mean absolute error for the same models reached a range from 2.81 minutes to 2.95 minutes. This means that models with only subsets of features did not have a large deterioration in performance compared to the model with full set of features. All models had a negative mean for residuals which indicates that the turnaround duration predictions are on average longer compared to actual turnarounds. However, largest positive residuals occurred for turnarounds with long actual turnaround durations. XGBoost model was demonstrated to have a better performance in predicting durations for turnarounds without delay codes compared to turnarounds with determined delays.

SHAP feature importance figure indicated that features MailIn and MailOut have almost non-existent importance values. Importance values for different aircraft variants did not demonstrate great significance either. Feature importance for the feature FuelNeed was low but according to the SHAP summary plot, FuelNeed has an effect of multiple minutes if fuelling is performed. Available TAT had a large feature importance but dependence plot for Available TAT proved that the effect of that feature on predictions is large only for large feature values. This means that when aircraft is late or just in schedule, the effect Available TAT is not that significant. According to feature importance values, largest effect on predictions comes from features PaxOnBoardOut and Deice. According to the SHAP summary plot, feature CargoIn has a negative SHAP value for large feature values of CargoIn. This means that turnarounds with incoming cargo have a shorter turnaround compared to turnarounds without incoming cargo. Reasoning behind such observation is not evident.

SHAP interaction values visualized with dependence plots indicate that duration for turnarounds with small departing passenger count is more affected by the deicing compared to turnarounds with larger number of departing passengers. For turnarounds with larger departing passenger count the SHAP value is lower when deicing is performed. This could be due to the property of the SHAP value calculation which leads to a situation where a value of a feature corresponding to the feature not being present creates a negative effect instead of a neutral effect on the output. Interaction with features PaxOnBoardOut and TimeOfDay_morning indicates, that during mornings turnarounds with a small passenger count have a longer duration. For higher number of passengers, the trend changes so that morning turnarounds have shorter durations and for turnarounds with highest number of passengers there is no difference whether it is morning or not. On the other hand, during middays turnarounds have a shorter duration with lower number of departing passengers, longer duration for higher number of passengers, and for highest number of passengers there is no distinct trend.

Dependence plot demonstrated that for available TAT, as discussed, the effect on prediction is significant only for large values of Available TAT. Turnaround duration when there is departing cargo is always shorter during middays. Features BagsIn and CargoIn demonstrated periodical upward and downward trends on SHAP values which could be due to a technical issue such as a cargo container size.

Example turnarounds from the test set were presented and five of those were furthermore analyzed with SHAP. The group of five turnaround predictions had one with a good prediction accuracy, two with the largest positive difference between actual and predicted turnaround durations and two with the largest negative difference. Based on the inspection of local feature effects, the common trend for largest positive residuals was that the actual turnaround times were high compared to scheduled turnarounds and the predictions were insufficient in predicting the magnitude of the actual duration. For turnarounds with largest negative residuals the common trend was that predictions overestimated the turnaround duration which in reality was not far away from the scheduled duration. Extreme feature values in the example cases with large errors can only partly explain the errors. Turnarounds, for which the prediction performance was good, did not have large differences between scheduled or actual turnaround durations.

All in all, as demonstrated already by the SHAP summary plot, inspection of individual predictions confirmed that in all of the cases only few features were meaningful. For cases with largest errors, on the other hand, a better performance might have required additional features to describe the phenomenon better.

6 Discussion and conclusion

The objective of this study was to create a proof-of-concept of a predictive model which could be used to both create interpretable predictions about individual turnarounds and to be able to describe a turnaround process based on explicit input parameters. In theory, a predictive model with sufficient amount of data together with methods to introduce interpretability for the predictions is able to reveal relationships in a turnaround process. The advantage of such an explainable aggregated approach is that there is no need to model individual subprocesses and to gather detailed information such as time stamp data.

A model was created by using the XGBoost machine learning algorithm together with the interpretability framework SHAP. The final model has a mean absolute error of 2.81 minutes and explained variance of 0.60. In 75 percent of the test cases the difference between actual turnaround durations and predictions was less than the difference between actual durations and the original schedule. In addition, the model is able to present the magnitude of the effect of individual features on a single prediction. These results indicate that from the perspective of predicting individual turnaround durations, the model can be used as an operative decision-making tool to approximate the duration and to identify the potential factors that may lead to a late departure for a single turnaround.

From the perspective of bringing transparency to turnarounds and identifying connections between variables in general, the model provides the following insights. First of all, only a subset of features in the data is sufficient in reaching a predictive accuracy that is close to the best accuracy presented above. In addition, the explained variance provided with the two best performing subsets is close to the explained variance above. This means that the number of departing passengers, amount of departing bags in kilograms, deicing, month of the turnaround, time of day, number of arriving passengers and number of departing special passengers are the main drivers of the turnaround duration. On the other hand, the amount of arriving or departing mail or the aircraft variant can be considered to have a negligible effect on the turnaround duration.

Secondly, there are no strong interactions between input features. However, winter season demonstrates a weak interaction with large amount of departing cargo, high number arriving of passengers and large amount of departing bags. The amount of departing bags interacts weakly with the number of departing special passengers. In addition, the effect of the number of departing passengers on the turnaround duration changes depending on the time of day and whether deicing is performed.

Based on these findings it is possible to conclude that boarding, which is illustrated through the passenger and special passenger numbers, is the most critical process due to its dominating effect on turnaround durations. Deicing process usually overlaps with the boarding process and its effect on the turnaround duration is partly dependent on the number of passengers. In general, ground operations which include the unloading and loading of bags and cargo, and which in this case are described through the amount of bags and cargo, are rarely on the critical path. Results indicate that during winter, especially with large amounts of bags, the effect of ground operations increases. Time of day partly determines what is the effect of the amount of departing cargo on the turnaround duration. However, based on the results, it is not possible to say in what cases cargo handling becomes the critical process, if ever.

From operative perspective the information discussed above helps in determining where to set the focus for further process development. By revealing the features that cause the majority of the effect on turnaround durations and by pointing out the situations when interaction between variables may cause changes to the criticality of subprocesses, this proof-of-concept model can be considered as a successful first step towards gaining better understanding of the turnaround process. In addition, the information on what is not known and what is not relevant is valuable as well.

The research questions were formulated in section 3 and are presented below:

1. *Which turnaround related parameters are needed to predict turnaround durations?*
2. *How are input parameters of a turnaround capable of explaining the turnaround process and provide interpretability?*

This study has demonstrated which of the available input parameters are needed to create predictions, what is the accuracy of such predictions and what portion of the variance in turnaround times such parameters can explain. As discussed, the accuracy of the model is on a level that it can be used to predict turnaround durations and therefore this research provides an answer for the research question 1.

The second research question was related to the interpretability methods and how they could provide understanding of the turnaround process. In this study the interpretability for turnaround durations was provided in two levels, global and local. Global interpretability revealed the importance of individual features for the whole dataset and also the most significant pairwise interactions between variables. Local feature effects demonstrated how different features affect the prediction of a single turnaround. This thesis has demonstrated that the main dependencies of a turnaround process can be deduced by combining the main trends from the global interpretability with the examples of local explanations. Such an approach can also be used to discover the potential special cases. Therefore, this study answers the research question 2.

6.1 Limitations of the study

In a scientific research such as this thesis, multiple choices have to be made including the selection of suitable research methods, the variables included in the dataset and the statistical tools and models used to achieve the objective of the study. Such choices result in an outcome that inspects the studied phenomenon from a certain perspective with a certain setup and also with certain limitations which should be acknowledged.

Current shortcomings of the interpretable predictive model can be summarized as follows. Firstly, as the predictions are based in most cases only on the effect of few variables, the effect of these variables is not sufficient when the actual turnaround duration is considerably longer than the scheduled duration. On the other hand, for turnarounds where the duration is shorter than expected, only few effective variables cause predictions to be too coarse and therefore the duration may be overestimated. Secondly, on average the predictive accuracy of the model is lower for turnarounds with a late departure and with a specified delay code. Both of these shortcomings are eventually consequences of lack of data; additional features are needed to describe important aspects of turnaround operations that are not yet included in the dataset.

Domain knowledge and availability of data have been major determinants in the selection of which variables have eventually been included in the dataset of this study. This inevitably leads to an outcome which may lack variables that have not either been considered as relevant or they are not easily available. As discussed, weather is one factor that is taken into account in this study only indirectly through the features Deicing and Month. However, by providing more detailed information related to weather such as possible precipitation or temperature, explainability of the model could be increased in a way that is not evident beforehand. On the other hand, information regarding the ground handling processes such as number of personnel on duty or information related to boarding processes inside the terminal and the possible variations in those processes are examples of data which could provide additional insights but was not available for this thesis.

6.2 Recommendations for future research

Based on the shortcomings, it is possible to derive prospects for future research. This study is based on a quantitative research approach which means that all the findings and conclusions are based on the dataset. As indicated by the results, dataset used in this thesis is not capable of explaining all the variation in the turnaround durations. Some issues such as the relationship between different times of day and the turnaround duration is illustrated only based on numerical values and therefore qualitative methods such as interviews could provide additional insight into the dependencies between variables.

Therefore, first proposal for future research is to search for additional variables that would increase the level of the explained variance. Such data could be gathered by performing interviews and other means of data collection for personnel working in airports and ground handling companies. Such groups could point out issues that they consider important regarding a successful and predictable turnaround process. Second proposal is to apply explainable aggregated approach for a different airport to create a benchmark analysis on what factors constitute a predictable turnaround in a different setting.

6.3 Scientific impact

The contribution of this study to the field of scientific knowledge is the novel approach called explainable aggregated approach to predict and explain durations of aircraft turnarounds without the need to explicitly model or measure individual subprocesses. Based on previous scientific research with comparison between task-based and aggregated approaches, this thesis furthermore specifies the comparison as task-based versus explainable aggregated approaches.

Task-based models utilize empirical explicit subprocess related data or simulated data that is based on empirically determined parameters. Explainable aggregated model, on the other hand, uses data which is assumed to describe different subprocess lengths to a sufficient extent. This means that task-based models have more accurate information related to subprocess durations whereas explainable aggregated models create estimates about subprocesses and their effects on the total turnaround duration.

The accuracy of task-based models comes with the requirement to have detailed data about subprocesses which might be difficult to obtain. Compared to task-based model, explainable aggregated model can discover patterns and create predictions with less detailed data. However, for explainable aggregated models it might be difficult to determine what kind of data is needed to discover all the wanted relationships in a turnaround. An explainable aggregated model requires data from several perspectives, as was shown in this study, and only appropriate data increases accuracy but the data required by an explainable aggregated model is different compared to data required by the task-based approach.

Therefore, both approaches benefit from additional data but the tradeoff between these two approaches is eventually about the type of the required data. Task-based approach requires detailed data which can be difficult to obtain whereas explainable aggregated approach can utilize more common data to infer effects of subprocesses on the turnaround. More common data such as information regarding passenger and material flows can be obtained more easily between different airports which also may result in a better scalability for the use of explainable aggregated models between airports. This study has been the first step with certain successes and evident shortcomings to demonstrate the use of explainable aggregated models and further research is required. Table 22 below illustrates the summary of benefits and disadvantages between these two approaches.

Table 22: Comparison of benefits and disadvantages between task-based and explainable aggregated approaches

	Explainable aggregated	**Task-based**
Benefits	- Requires less detailed subprocess data - Easier to scale to other airports - Better data availability between airports	- Provides accurate empirical information related to subprocess durations
Disadvantages	- Relationships between variables have to be derived from the model - Might be hard to determine all the relevant variables	- Need for detailed subprocess related data - Data collection may be difficult - Laborious to scale for other airports

References

Airbus. (2020). *A320 Aircraft Characteristics for Airport and Maintenance Planning*. Airbus. Retrieved May 20, 2020, from https://www.airbus.com/aircraft/support-services/airport-operations-and-technical-data/aircraft-characteristics.html

Antoniou, A. (1992). The Factors Determining the Profitability of International Airlines: Some Econometric Results. *Managerial and Decision Economics, 13*(6), 503-514. http://doi.org/10.1002/mde.4090130606

Assaia. (n.d.). *Assaia Apron AI*. Assaia. Retrieved May 30, 2020, from https://assaia.com

Bishop, C. M. (2006). *Pattern recognition and machine learning*. Springer.

Boeing. (2020). *737 Airplane Characteristics for Airport Planning*. Boeing. Retrieved May 30, 2020, from https://www.boeing.com/commercial/airports/plan_manuals.page

Bouneder, L., Léo, Y., & Lachapelle, A. (2020). *X-SHAP: towards multiplicative explainability of Machine Learning*. arXiv. https://arxiv.org/abs/2006.04574

Bzdok, D., Altman, N., & Krzywinski, M. (2018). Points of significance: Statistics versus machine learning. *Nature Methods, 15*(4), 233-234. http://doi.org/10.1038/nmeth.4642

Chakki. (2017). PDPbox (Version 0.2.0) [Github repository]. Retrieved from https://github.com/SauceCat/PDPbox

Chen, J., & Li, M. (2019). Chained predictions of flight delay using machine learning. *AIAA Scitech 2019 Forum*, 1661.

Chen, T., & Guestrin, C. (2016). *XGBoost: A scalable tree boosting system.* arXiv. https://arxiv.org/abs/1603.02754

Chen, T., He, T., Benesty, M., & Tang, Y. (n.d.). *Understand Your Dataset with Xgboost*. CRAN. Retrieved June 13, 2020, from https://cran.r-project.org/web/packages/xgboost/vignettes/discoverYourData.html#numeric-v.s.-categorical-variables

Choi, S., Kim, Y. J., Briceno, S., & Mavris, D. (2017, September). Cost-sensitive prediction of airline delays using machine learning. *2017 IEEE/AIAA 36th Digital Avionics Systems Conference (DASC),* 1-8. https://doi.org/10.1109/DASC.2017.8102035

Clarke, M. D. D. (1998). Irregular airline operations: a review of the state-of-the-practice in airline operations control centers. *Journal of Air Transport Management*, *4*(2), 67-76. https://doi.org/10.1016/S0969-6997(98)00012-X

Cortina, J. M. (1993). Interaction, nonlinearity, and multicollinearity: Implications for multiple regression. *Journal of Management*, *19*(4), 915-922. https://doi.org/10.1016/0149-2063(93)90035-L

Dhaliwal, S. S., Nahid, A. A., & Abbas, R. (2018). Effective intrusion detection system using XGBoost. *Information*, *9*(7), 149. https://doi.org/10.3390/info9070149

Daoud, J. I. (2017, December). Multicollinearity and regression analysis. In *Journal of Physics: Conference Series*, *949*(1), 012009. https://doi.org/10.1088/1742-6596/949/1/012009

Dash, M., & Liu, H. (1997). Feature selection for classification. *Intelligent data analysis*, *1*(3), 131-156. https://doi.org/10.3233/IDA-1997-1302

Dawson, R. (2011). How significant is a boxplot outlier? *Journal of Statistics Education*, *19*(2). https://doi.org/10.1080/10691898.2011.11889610

Diepen, G., Pieters, B. F. I., Van Den Akker, J. M., & Hoogeveen, J. A. (2013). Robust planning of airport platform buses. *Computers & operations research*, *40*(3), 747-757. https://doi.org/10.1016/j.cor.2011.08.002

Doshi-Velez, F., & Kim, B. (2017). *Towards a rigorous science of interpretable machine learning*. arXiv. https://arxiv.org/abs/1702.08608

Eurocontrol. (2018, August 3). *Delays – three questions and many answers*. Retrieved June 10, 2020, from https://www.eurocontrol.int/news/delays-three-questions-and-many-answers

Eurocontrol. (2020, April 3). *CODA Digest: All-causes delay and cancellations to Air Transport in Europe for 2019*. Retrieved June 13, 2020, from https://www.eurocontrol.int/publication/all-causes-delay-and-cancellations-air-transport-europe-2019

Fricke, H., & Schultz, M. (2009, June). Delay impacts onto turnaround performance. *ATM Seminar*.

Friedman, J. H., & Popescu, B. E. (2008). Predictive learning via rule ensembles. *The Annals of Applied Statistics*, *2*(3), 916-954. https://doi.org/10.1214/07-AOAS148

Friedman, J., Hastie, T., & Tibshirani, R. (2001). *The elements of statistical learning*. Springer. https://doi.org/10.1007/978-0-387-84858-7

Gao, Y., Huyan, Z., & Ju, F. (2015, December). A Prediction Method Based on Neural Network for Flight Turnaround Time at Airport. *2015 8th International Symposium on Computational Intelligence and Design (ISCID)*, 2, 219-222. https://doi.org/10.1109/ISCID.2015.44

Gilpin, L.H., Bau, D., Yuan, B., Bajwa, A., Specter, M., & Kagal, L. (2018). Explaining Explanations: An Overview of Interpretability of Machine Learning. *2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA)*, 80-89. https://doi.org/10.1109/DSAA.2018.00018

Goldstein, A., Kapelner, A., Bleich, J., & Pitkin, E. (2015). Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation. *Journal of Computational and Graphical Statistics*, *24*(1), 44-65. https://doi.org/10.1080/10618600.2014.907095

Granitto, P. M., Furlanello, C., Biasioli, F., & Gasperi, F. (2006). Recursive feature elimination with random forest for PTR-MS analysis of agroindustrial products. *Chemometrics and Intelligent Laboratory Systems*, *83*(2), 83-90. https://doi.org/10.1016/j.chemolab.2006.01.007

Guerreiro Fregnani, J. A. T., Müller, C., & Correia, A. R. (2013). A fuel tankering model applied to a domestic airline network. *Journal of Advanced Transportation*, *47*(4), 386-398. https://doi.org/10.1002/atr.162

Guyon, I., & Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of machine learning research*, *3*(7/8), 1157-1182. https://doi.org/ 10.5555/944919.944968

Hall, P. (2018). *On the art and science of machine learning explanations*. arXiv. https://arxiv.org/abs/1810.02909

Hall, P., & Gill, N. (2019). *An introduction to machine learning interpretability*. O'Reilly Media.

Hassel van, O. F. J. (2019). *Predicting the turnaround time of an aircraft: a process structure aware approach*. [Master's thesis, Eindhoven University of Technology]. Retrieved May 10, 2020, from https://research.tue.nl/en/studentTheses/predicting-the-turnaround-time-of-an-aircraft

Haygood, R. (2017). sklearn-gbmi (Version 1.0.3) [Github repository]. Retrieved from https://github.com/ralphhaygood/sklearn-gbmi

Horiguchi, Y., Baba, Y., Kashima, H., Suzuki, M., Kayahara, H., & Maeno, J. (2017, February). Predicting fuel consumption and flight delays for low-cost airlines. *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, 4686-4693. https://doi.org/10.5555/3297863.3297874

Iglewicz, B., & Hoaglin, D. C. (1993). *How to detect and handle outliers*. Asq Press.

Jaehn, F., & Neumann, S. (2015). Airplane boarding. *European Journal of Operational Research*, *244*(2), 339-359. https://doi.org/10.1016/j.ejor.2014.12.008

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An introduction to statistical learning*. Springer. https://doi.org/10.1007/978-1-4614-7138-7

Jordan, M. I., & Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects. *Science*, *349*(6245), 255-260. https://doi.org/10.1126/science.aaa8415

Kohl, N., Larsen, A., Larsen, J., Ross, A., & Tiourine, S. (2007). Airline disruption management—perspectives, experiences and outlook. *Journal of Air Transport Management*, *13*(3), 149-162. https://doi.org/10.1016/j.jairtraman.2007.01.001

Kuhn, M., & Johnson, K. (2013). *Applied predictive modeling*. Springer. https://doi.org/10.1007/978-1-4614-6849-3

Kwak, S. K., & Kim, J. H. (2017). Statistical data preparation: management of missing values and outliers. *Korean journal of anesthesiology*, *70*(4), 407. https://doi.org/10.4097/kjae.2017.70.4.407

Lawton, T. C. (2003). Managing proactively in turbulent times: Insights from the low-fare airline business. *Irish Journal of Management*, *24*(1), 173. https://doi.org/10.4324/9781315242323-11

Lipovetsky, S., & Conklin, M. (2001). Analysis of regression in game theory approach. *Applied Stochastic Models in Business and Industry*, *17*(4), 319-330. https://doi.org/10.1002/asmb.446

Lohatepanont, M., & Barnhart, C. (2004). Airline schedule planning: Integrated models and algorithms for schedule design and fleet assignment. *Transportation science*, *38*(1), 19-32. http://doi.org/10.1287/trsc.1030.0026

Lundberg, S. (2018). SHAP (Version 0.36.0) [Github repository]. Retrieved from https://github.com/slundberg/shap

Lundberg, S. M., & Lee, S. I. (2017). A unified approach to interpreting model predictions. *Advances in neural information processing systems,* 4765-4774. arXiv. https://arxiv.org/abs/1705.07874

Lundberg, S. M., Erion, G. G., & Lee, S. I. (2018). *Consistent individualized feature attribution for tree ensembles*. arXiv. https://arxiv.org/abs/1802.03888

Lundberg, S. M., Erion, G., Chen, H., DeGrave, A., Prutkin, J. M., Nair, B., Katz, R., Himmelfarb, J., Bansal, N., & Lee, S. I. (2020). From local explanations to global understanding with explainable AI for trees. *Nature machine intelligence*, *2*(1), 2522-5839. https://doi.org/10.1038/s42256-019-0138-9

Manna, S., Biswas, S., Kundu, R., Rakshit, S., Gupta, P., & Barman, S. (2017, June). A statistical approach to predict flight delay using gradient boosted decision tree. *2017 International Conference on Computational Intelligence in Data Science (ICCIDS),* 1-5. https://doi.org/10.1109/ICCIDS.2017.8272656

Mansfield, E. R., & Helms, B. P. (1982). Detecting multicollinearity. *The American Statistician*, *36*(3), 158-160. https://doi.org/10.2307/2683167

Mitchell, R., & Frank, E. (2017). Accelerating the XGBoost algorithm using GPU computing. *PeerJ Computer Science*, *3*, e127. https://doi.org/10.7717/peerj-cs.127

Molnar, C. (2019). *Interpretable Machine Learning*. Interpretable Machine Learning. Retrieved June 30, 2020, from https://christophm.github.io/interpretable-ml-book/

Murphy, K. P. (2012). *Machine learning: a probabilistic perspective*. MIT press.

Nagelkerke, N. J. (1991). A note on a general definition of the coefficient of determination. *Biometrika*, *78*(3), 691-692. https://doi.org/10.2307/2337038

Norin, A., Granberg, T. A., Yuan, D., & Värbrand, P. (2012). Airport logistics–A case study of the turn-around process. *Journal of Air Transport Management*, *20*, 31-34. https://doi.org/10.1016/j.jairtraman.2011.10.008

Nosedal Sanchez, J., & Piera Eroles, M. A. (2018). Causal analysis of aircraft turnaround time for process reliability evaluation and disruptions' identification. *Transportmetrica B: Transport Dynamics*, *6*(2), 115-128. https://doi.org/10.1080/21680566.2017.1325784

Oreschko, B., Schultz, M., Elflein, J., & Fricke, H. (2010). Significant Turnaround Process Variations due to Airport. *Air Transport and Operations: Proceedings of the First International Air Transport and Operations Symposium 2010*, 263.

Oreschko, B., Kunze, T., Schultz, M., Fricke, H., Kumar, V., & Sherry, L. (2012, May). Turnaround prediction with stochastic process times and airport specific delay pattern. *International Conference on Research in Airport Transportation (ICRAT), Berkeley*.

Pandas (Version 1.1.0) [Python library]. (2020). Retrieved from https://pandas.pydata.org/

Pareto, V. (1896). *Cours d'économie politique: professé à l'Universįté de Lausanne*. F. Rouge.

Regulation 859/2008. (2008). *OPS 1.305: Refuelling/defuelling with passengers embarking, on board or disembarking*. European Union, European Commission. Retrieved from https://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:2008:254:0001:0238:En:PDF

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016, August). " Why should I trust you?" Explaining the predictions of any classifier. *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 1135-1144. https://doi.org/10.1145/2939672.2939778

San Antonio, A., Juan, A. A., Calvet, L., i Casas, P. F., & Guimarans, D. (2017, December). Using simulation to estimate critical paths and survival functions in aircraft turnaround processes. *2017 Winter*

Simulation Conference (WSC), 3394-3403. https://doi.org/10.1109/WSC.2017.8248055

Schmidt, M. (2017). A review of aircraft turnaround operations and simulations. *Progress in Aerospace Sciences*, *92*, 25-38. https://doi.org/10.1016/j.paerosci.2017.05.002

Schultz, M., & Reitmann, S. (2019). Machine learning approach to predict aircraft boarding. *Transportation Research Part C: Emerging Technologies*, *98*, 391-408. https://doi.org/10.1016/j.trc.2018.09.007

Scikit-learn (Version 0.23) [Python library]. (2020). Retrieved from https://scikit-learn.org/stable/

Schober, P., Boer, C., & Schwarte, L. A. (2018). Correlation coefficients: appropriate use and interpretation. *Anesthesia & Analgesia*, *126*(5), 1763-1768. https://doi.org/10.1213/ANE.0000000000002864

Tukey, J.W., 1977. *Exploratory data analysis*. Addison-Wesley.

Vapnik, V. N. (1999). An overview of statistical learning theory. *IEEE transactions on neural networks*, *10*(5), 988-999. https://doi.org/10.1109/72.788640

Wainer, J., & Cawley, G. (2018). *Nested cross-validation when selecting classifiers is overzealous for most practical applications*. arXiv. https://arxiv.org/abs/1809.09446

Wan, C., Zhaoxin, C., Yuxian, D., Yang, L., Ye, P., & Tao, H. (2019, October). Dynamic Prediction about Turnaround Time of Flight based on Support Vector Machine Regression. In *2019 IEEE 1st International Conference on Civil Aviation Safety and Information Technology (ICCASIT)*, 109-113. https://doi.org/10.1109/ICCASIT48058.2019.8973132

Wang, H., Wang, M., & Wu, Y. (2017, August). Development of an Aircraft Turnaround Time Estimation Model Based on Discrete Time Simulation. *ITITS,* 29-36. https://doi.org/10.3233/978-1-61499-785-6-29

Willmott, C. J., & Matsuura, K. (2005). Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. *Climate research*, *30*(1), 79-82. https://doi.org/10.3354/cr030079

Wohlin, C., Höst, M., & Henningsson, K. (2003). Empirical research methods in software engineering. *Empirical methods and studies in software engineering*, 7-23. https://doi.org/10.1007/978-3-540-45143-3_2

Wu, C. L. (2008). Monitoring aircraft turnaround operations–framework development, application and implications for airline operations. *Transportation Planning and Technology*, *31*(2), 215-228. https://doi.org/10.1080/03081060801948233

Wu, C.L., 2010. *Airline operations and delay management: insights from airline economics, networks and strategic schedule planning*. Ashgate. https://doi.org/10.4324/9781315566467

Wu, C. L., & Caves, R. E. (2004a). Modelling and optimization of aircraft turnaround time at an airport. *Transportation Planning and Technology*, *27*(1), 47-66. https://doi.org/10.1080/0308106042000184454

Wu, C. L., & Caves, R. E. (2004b). Modelling and simulation of aircraft turnaround operations at airports. *Transportation Planning and Technology*, *27*(1), 25-46. https://doi.org/10.1080/0308106042000184445

Xgboost developers. (2020). *XGBoost documentation*. Retrieved July 1, 2020, from https://xgboost.readthedocs.io/en/latest/index.html

zeroG. (n.d.). *Deep Turnaround*. zeroG. Retrieved May 30, 2020, from https://www.zerog.aero/solutions/deep-turnaround/

Zhang, D., Qian, L., Mao, B., Huang, C., Huang, B., & Si, Y. (2018). A data-driven design for fault detection of wind turbines using random forests and XGboost. *IEEE Access*, *6*, 21020-21031. https://doi.org/10.1109/ACCESS.2018.2818678

A Derivation of the XGBoost algorithm

In this part the derivation of the needed formulas for the XGBoost is performed. According to Chen and Guestrin (2016), the objective function

$$\mathcal{L}^{(t)} = \sum_{i=1}^{n} l(y_i, \hat{y}_i^{(t-1)} + f_t(\mathbf{x}_i)) + \Omega(f_t) \tag{A1}$$

where $\Omega(f) = \gamma T + \frac{1}{2}\lambda \parallel w \parallel^2$

needs to be transformed to an approximated form to be optimized in Euclidean space. The main idea is to transform Equation A1 to a form that can be used to obtain optimal outputs from the regression trees. Symbols presented by Equation A1 have been discussed already in section 4.3.2. The derivation of different formulas presented by Chen and Guestrin (2016) is shown below. Second-order Taylor approximation of the objective function is presented in form

$$\mathcal{L}^{(t)} \simeq \sum_{i=1}^{n} [l\left(y_i, \hat{y}^{(t-1)}\right) + g_i f_t(\mathbf{x}_i) + \frac{1}{2} h_i f_t^2(\mathbf{x}_i)] + \Omega(f_t) \tag{A2}$$

where g_i is the gradient, or the first derivative, of the loss function, and h_i is the Hessian, or the second derivate, of the loss function. For optimization purposes the removal of constants of Equation A2 results in form

$$\tilde{\mathcal{L}}^{(t)} = \sum_{j=1}^{T} [G_j w_j + \frac{1}{2}(H_j + \lambda) w_j^2] + \gamma T\ . \tag{A3}$$

In addition to removal of constants in Equation A3, symbols G_j and H_j represent the sums of derivatives in leaf j. Symbol w_j, on the other hand, corresponds to the weights of leaf j. This presentation of sums of derivatives and leaf weights follows the notation presented by Mitchell and Frank (2017).

By taking derivative of $\tilde{\mathcal{L}}$ with respect to w_j and by calculating when the derivative is zero, it results in the optimal weight, or output, of leaf j. This optimal value is presented as

$$w_j^* = -\frac{G_j}{H_j + \lambda}. \tag{A4}$$

When optimal leaf weight presented by Equation A4 is inserted into Equation A3, it results in objective function which provides the best tree structure in form

$$\tilde{\mathcal{L}}^{(t)} = -\frac{1}{2} \sum_{j=1}^{T} \frac{G_j^2}{H_j + \lambda} + \gamma T. \tag{A5}$$

From Equation A5 it is possible to derive the gain function that is used to evaluate the splits in a tree as originally presented by Chen and Guestrin (2016). The gain function presented as

$$Gain = \frac{1}{2}\left[\frac{G_L^2}{H_L+\lambda} + \frac{G_R^2}{H_R+\lambda} - \frac{(G_L+G_R)^2}{H_L+H_R+\lambda}\right] - \gamma \tag{A6}$$

calculates the improvement in the objective function when a leaf is split into two new leaves. New leaves left and right are indicated with index notations L and R whereas $L + R$ corresponds to a leaf without splitting. The default loss function of XGBoost is the mean squared error loss which is expressed as

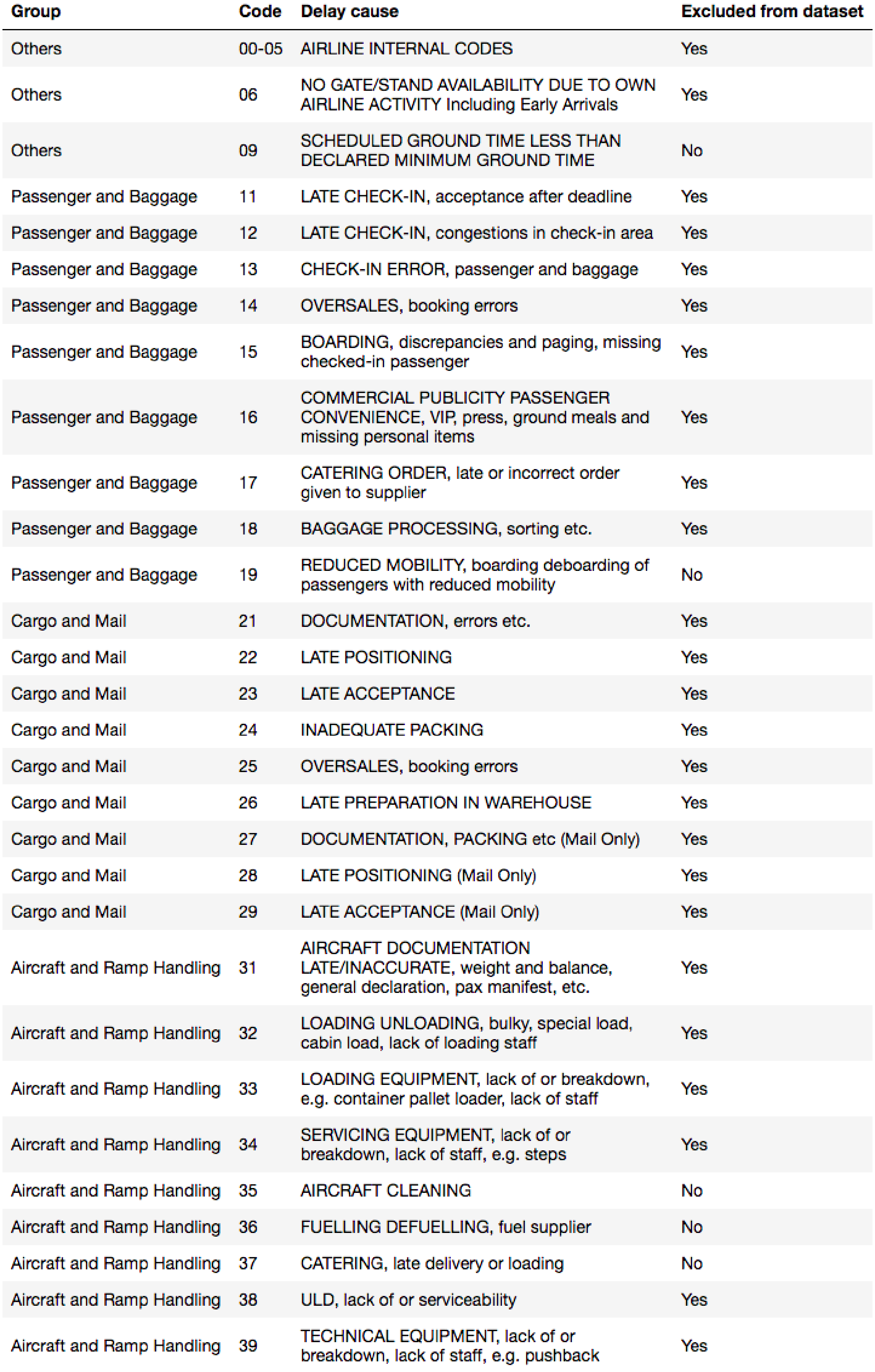
$$l(y, \hat{y}) = \frac{1}{2}(y - \hat{y})^2. \tag{A7}$$

Equations A4, A5, A6 and the loss function presented by Equation A7 represent the main functions used in the XGBoost algorithm.

B List of delay codes used in preprocessing

This appendix presents three tables of delay codes. Column "Excluded from dataset" in Table 23, Table 24 and Table 25 below indicates whether a specific delay code has been excluded from the input data. As discussed in earlier sections, the reason for filtering out certain delay codes is due to the reason that such delays cannot be deduced from the dataset. Delay code data is provided by Eurocontrol (2020) and they are based on the standard IATA delay codes.

Table 23: Delay codes from 00 to 39

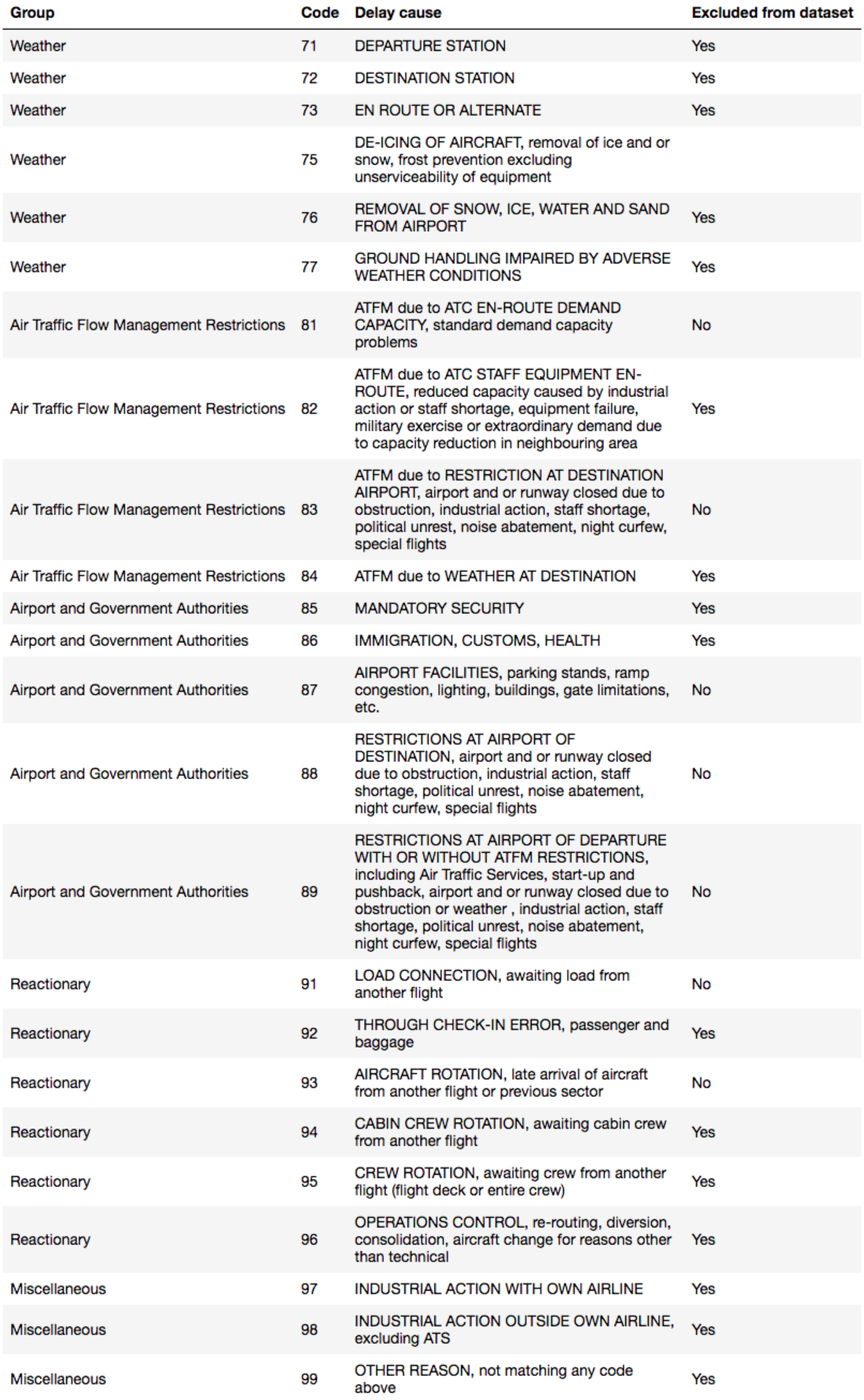


Group	Code	Delay cause	Excluded from dataset
Others	00-05	AIRLINE INTERNAL CODES	Yes
Others	06	NO GATE/STAND AVAILABILITY DUE TO OWN AIRLINE ACTIVITY Including Early Arrivals	Yes
Others	09	SCHEDULED GROUND TIME LESS THAN DECLARED MINIMUM GROUND TIME	No
Passenger and Baggage	11	LATE CHECK-IN, acceptance after deadline	Yes
Passenger and Baggage	12	LATE CHECK-IN, congestions in check-in area	Yes
Passenger and Baggage	13	CHECK-IN ERROR, passenger and baggage	Yes
Passenger and Baggage	14	OVERSALES, booking errors	Yes
Passenger and Baggage	15	BOARDING, discrepancies and paging, missing checked-in passenger	Yes
Passenger and Baggage	16	COMMERCIAL PUBLICITY PASSENGER CONVENIENCE, VIP, press, ground meals and missing personal items	Yes
Passenger and Baggage	17	CATERING ORDER, late or incorrect order given to supplier	Yes
Passenger and Baggage	18	BAGGAGE PROCESSING, sorting etc.	Yes
Passenger and Baggage	19	REDUCED MOBILITY, boarding deboarding of passengers with reduced mobility	No
Cargo and Mail	21	DOCUMENTATION, errors etc.	Yes
Cargo and Mail	22	LATE POSITIONING	Yes
Cargo and Mail	23	LATE ACCEPTANCE	Yes
Cargo and Mail	24	INADEQUATE PACKING	Yes
Cargo and Mail	25	OVERSALES, booking errors	Yes
Cargo and Mail	26	LATE PREPARATION IN WAREHOUSE	Yes
Cargo and Mail	27	DOCUMENTATION, PACKING etc (Mail Only)	Yes
Cargo and Mail	28	LATE POSITIONING (Mail Only)	Yes
Cargo and Mail	29	LATE ACCEPTANCE (Mail Only)	Yes
Aircraft and Ramp Handling	31	AIRCRAFT DOCUMENTATION LATE/INACCURATE, weight and balance, general declaration, pax manifest, etc.	Yes
Aircraft and Ramp Handling	32	LOADING UNLOADING, bulky, special load, cabin load, lack of loading staff	Yes
Aircraft and Ramp Handling	33	LOADING EQUIPMENT, lack of or breakdown, e.g. container pallet loader, lack of staff	Yes
Aircraft and Ramp Handling	34	SERVICING EQUIPMENT, lack of or breakdown, lack of staff, e.g. steps	Yes
Aircraft and Ramp Handling	35	AIRCRAFT CLEANING	No
Aircraft and Ramp Handling	36	FUELLING DEFUELLING, fuel supplier	No
Aircraft and Ramp Handling	37	CATERING, late delivery or loading	No
Aircraft and Ramp Handling	38	ULD, lack of or serviceability	Yes
Aircraft and Ramp Handling	39	TECHNICAL EQUIPMENT, lack of or breakdown, lack of staff, e.g. pushback	Yes

Table 24: Delay codes from 41 to 69

Group	Code	Delay cause	Excluded from dataset
Technical and Aircraft Equipment	41	AIRCRAFT DEFECTS	Yes
Technical and Aircraft Equipment	42	SCHEDULED MAINTENANCE, late release	Yes
Technical and Aircraft Equipment	43	NON-SCHEDULED MAINTENANCE, special checks and or additional works beyond normal maintenance schedule	Yes
Technical and Aircraft Equipment	44	SPARES AND MAINTENANCE EQUIPMENT, lack of or breakdown	Yes
Technical and Aircraft Equipment	45	AOG SPARES, to be carried to another station	Yes
Technical and Aircraft Equipment	46	AIRCRAFT CHANGE, for technical reasons	Yes
Technical and Aircraft Equipment	47	STAND-BY AIRCRAFT, lack of planned stand-by aircraft for technical reasons	Yes
Technical and Aircraft Equipment	48	SCHEDULED CABIN CONFIGURATION VERSION ADJUSTMENTS	Yes
Damage to Aircraft & EDP Automated Equipment Failure	51	DAMAGE DURING FLIGHT OPERATIONS, bird or lightning strike, turbulence, heavy or overweight landing, collision during taxiing	Yes
Damage to Aircraft & EDP Automated Equipment Failure	52	DAMAGE DURING GROUND OPERATIONS, collisions (other than during taxiing), loading off-loading damage, contamination, towing, extreme weather conditions	Yes
Damage to Aircraft & EDP Automated Equipment Failure	55	DEPARTURE CONTROL	Yes
Damage to Aircraft & EDP Automated Equipment Failure	56	CARGO PREPARATION DOCUMENTATION	Yes
Damage to Aircraft & EDP Automated Equipment Failure	57	FLIGHT PLANS	Yes
Damage to Aircraft & EDP Automated Equipment Failure	58	OTHER AUTOMATED SYSTEM	Yes
Flight Operations and Crewing	61	FLIGHT PLAN, late completion or change of, flight documentation	Yes
Flight Operations and Crewing	62	OPERATIONAL REQUIREMENTS, fuel, load alteration	No
Flight Operations and Crewing	63	LATE CREW BOARDING OR DEPARTURE PROCEDURES, other than connection and standby (flight deck or entire crew)	Yes
Flight Operations and Crewing	64	FLIGHT DECK CREW SHORTAGE, sickness, awaiting standby, flight time limitations, crew meals, valid visa, health documents, etc.	Yes
Flight Operations and Crewing	65	FLIGHT DECK CREW SPECIAL REQUEST, not within operational requirements	Yes
Flight Operations and Crewing	66	LATE CABIN CREW BOARDING OR DEPARTURE PROCEDURES, other than connection and standby	Yes
Flight Operations and Crewing	67	CABIN CREW SHORTAGE, sickness, awaiting standby, flight time limitations, crew meals, valid visa, health documents, etc.	Yes
Flight Operations and Crewing	68	CABIN CREW ERROR OR SPECIAL REQUEST, not within operational requirements	Yes
Flight Operations and Crewing	69	CAPTAIN REQUEST FOR SECURITY CHECK, extraordinary	Yes

Table 25: Delay codes from 71 to 99



Group	Code	Delay cause	Excluded from dataset
Weather	71	DEPARTURE STATION	Yes
Weather	72	DESTINATION STATION	Yes
Weather	73	EN ROUTE OR ALTERNATE	Yes
Weather	75	DE-ICING OF AIRCRAFT, removal of ice and or snow, frost prevention excluding unserviceability of equipment	
Weather	76	REMOVAL OF SNOW, ICE, WATER AND SAND FROM AIRPORT	Yes
Weather	77	GROUND HANDLING IMPAIRED BY ADVERSE WEATHER CONDITIONS	Yes
Air Traffic Flow Management Restrictions	81	ATFM due to ATC EN-ROUTE DEMAND CAPACITY, standard demand capacity problems	No
Air Traffic Flow Management Restrictions	82	ATFM due to ATC STAFF EQUIPMENT EN-ROUTE, reduced capacity caused by industrial action or staff shortage, equipment failure, military exercise or extraordinary demand due to capacity reduction in neighbouring area	Yes
Air Traffic Flow Management Restrictions	83	ATFM due to RESTRICTION AT DESTINATION AIRPORT, airport and or runway closed due to obstruction, industrial action, staff shortage, political unrest, noise abatement, night curfew, special flights	No
Air Traffic Flow Management Restrictions	84	ATFM due to WEATHER AT DESTINATION	Yes
Airport and Government Authorities	85	MANDATORY SECURITY	Yes
Airport and Government Authorities	86	IMMIGRATION, CUSTOMS, HEALTH	Yes
Airport and Government Authorities	87	AIRPORT FACILITIES, parking stands, ramp congestion, lighting, buildings, gate limitations, etc.	No
Airport and Government Authorities	88	RESTRICTIONS AT AIRPORT OF DESTINATION, airport and or runway closed due to obstruction, industrial action, staff shortage, political unrest, noise abatement, night curfew, special flights	No
Airport and Government Authorities	89	RESTRICTIONS AT AIRPORT OF DEPARTURE WITH OR WITHOUT ATFM RESTRICTIONS, including Air Traffic Services, start-up and pushback, airport and or runway closed due to obstruction or weather , industrial action, staff shortage, political unrest, noise abatement, night curfew, special flights	No
Reactionary	91	LOAD CONNECTION, awaiting load from another flight	No
Reactionary	92	THROUGH CHECK-IN ERROR, passenger and baggage	Yes
Reactionary	93	AIRCRAFT ROTATION, late arrival of aircraft from another flight or previous sector	No
Reactionary	94	CABIN CREW ROTATION, awaiting cabin crew from another flight	Yes
Reactionary	95	CREW ROTATION, awaiting crew from another flight (flight deck or entire crew)	Yes
Reactionary	96	OPERATIONS CONTROL, re-routing, diversion, consolidation, aircraft change for reasons other than technical	Yes
Miscellaneous	97	INDUSTRIAL ACTION WITH OWN AIRLINE	Yes
Miscellaneous	98	INDUSTRIAL ACTION OUTSIDE OWN AIRLINE, excluding ATS	Yes
Miscellaneous	99	OTHER REASON, not matching any code above	Yes