Predicting Runway Allocation with Support Vector Machine and Logistic Regression

Kia Eisinga ANR: 707043

Thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in Communication and Information Sciences, Master Track Data Science: Business and Governance, at the School of Humanities of Tilburg University

Thesis committee:

dr. Grzegorz A. Chrupała dr.ing. Sander C. J. Bakkes

Tilburg University School of Humanities Department of Communication and Information Sciences Tilburg center for Cognition and Communication (TiCC) Tilburg, The Netherlands July 2016

Contents

1	Intr	roduction 1
	1.1	Context
		1.1.1 Academic Relevance
		1.1.2 Practical Relevance
	1.2	Research Questions
	1.3	Structure
2	\mathbf{Rel}	ated Work 6
	2.1	Previous Work
		2.1.1 Optimisation $\ldots \ldots 7$
		2.1.2 Heuristics
		2.1.3 Discrete-choice Models
		2.1.4 Neural Networks
	2.2	Research Gaps 11
	2.3	Current Study 12
3	Exp	perimental Setup 15
	3.1	Dataset
		3.1.1 Exploratory Data Analysis
	3.2	Cleaning and Pre-processing
		3.2.1 Missing Values and Feature Extraction
		3.2.2 Outliers
		3.2.3 Feature Engineering
	3.3	Experimental Procedure
		3.3.1 Parameter Grid Search
		3.3.2 Feature Influence
	3.4	Evaluation Criteria
4	_	
_	Res	ults 28

	4.2	Baseline	29
	4.3	Precision, Recall and F_1 -score	30
		4.3.1 Per-class Performance	31
5	Disc	cussion and Conclusion	33
	5.1	Limitations	34
	5.2	Contribution to the Existing Framework	35
	5.3	Future Research	36
	5.4	Implications for the Field	37
Ap	pen	dices	40
	Ā	Dataset description	40
	В	Results ablation study	44
	С	Confusion matrices	46

List of Figures

3.1	Runway map of Amsterdam Airport Schiphol	16
3.2	Air traffic growth at Amsterdam Airport Schiphol	17
3.3	Total number of flights for each runway	18
3.4	Total number of light airplane flights ($\leq 7,000$ kg) for each	
	runway	19
3.5	Arrivals and departures on each runway	19
3.5 3.6	Arrivals and departures on each runway	19 20

List of Tables

3.1	Optimal parameters found by grid search	25
3.2	CA results with linear kernel SVM when omitting feature	
	subsets	26
3.3	CA results with Logistic Regression when omitting feature	
	subsets	27
4.1	CA after testing on the test set	28
4.2	Number of flights per runway	29
4.3	Average performance scores on test set	30
4.4	Precision, Recall and F_1 -scores for linear kernel SVM	31
4.5	Precision, Recall and F_1 -scores for Logistic Regression	32
B.1	CA results for ablation study with linear kernel SVM	44
B.2	CA results for ablation study with Logistic Regression	45
C.1	Confusion matrix for the linear kernel SVM classifier	46
C.2	Confusion matrix for the Logistic Regression classifier	47

Acronyms

- **AAS** Amsterdam Airport Schiphol. 1–4, 6, 7, 12–18, 20–23, 29, 33, 34, 36, 37
- ATC Air Traffic Control. 1–3, 6–10, 13, 14, 17, 18, 23, 27, 28, 30, 33–37
- BAS Bezoekers Aanspreekpunt Schiphol. 15, 20, 21
- CA Classification Accuracy. 4, 5, 14, 25–30, 34, 36, 44
- EDA Exploratory Data Analysis. 17, 21, 23, 33
- ICAO International Civil Aviation Organisation. 23, 40
- ILS Instrumental Landing Systems. 2, 23, 26
- IMC Instrumental Meteorological Conditions. 23, 42
- KNMI Royal Netherlands Meteorological Institute. 15, 20, 21
- SGD Stochastic Gradient Descent. 2, 3, 8, 11, 24, 25, 28, 35
- SVM Support Vector Machine. 4, 5, 11–14, 25, 26, 28, 31, 34–36, 46
- VMC Visual Meteorological Conditions. 23, 42

Acknowledgements

This thesis concludes my fulfillment of the Master of Science in Communication and Information Sciences, master track Data Science: Business and Governance. I would like to express special gratitude to a number of people who have helped me along the process of writing this thesis.

First and foremost, I would like to say many thanks to my supervisor Grzegorz Chrupała, who has helped me with his expertise and willingness to answer any question I could have. It was great to get the chance to work with you and learn from you.

Another thanks goes to Marie Postma and Jeroen Janssens from Tilburg University for additional guidance on my thesis. Thank you Marie for helping me with making the right decisions in the early process of this thesis, and thank you Jeroen for your additional classes and help in R.

A very important person I would like to thank is my supervisor from Heijmans, Jordi Overeem. Jordi and I have held frequent brain-storming sessions throughout the entire process of writing my thesis, all of which helped me get to this point. Your patience and optimism were a true benefit.

Another person who has helped me with her fair share of brain-storming is Nynke van der Lei. In the role of management trainee at Heijmans, she helped me get access to numerous data sources available at Heijmans and brought forth a lot of ideas on which I could build upon. It was a pleasure to work with you.

I would also like to say a big thank you to Gerard de Leede, CTO at Heijmans, who gave me the opportunity to combine my master thesis with a graduate internship at Heijmans. Apart from my academic growth, it was wonderful to gain professional experience at such an ambitious and innovative company.

A special round of thanks goes to Michel Vermeij from KLM and Brian Ho-Sam-Sooi from Amsterdam Airport Schiphol. Although we have not met personally, you were kind enough to share some of your knowledge and resources with me. Thank you for your benevolence.

Finally, I would like to thank my parents, Francis Stortelder and Rob Eisinga, who have helped me with their ample advise and expertise for many years. Thank you!

Abstract

As air transportation continues to grow and expanding the number of runways is often impractical, it has become increasingly important for Air Traffic Control (ATC) to allocate runways more efficiently. The current study focuses on predicting runway allocation for all arriving and departing flights at Amsterdam Airport Schiphol, to enhance the capacity of ATC.

Previous research has included optimisation techniques, a heuristics approach, discrete-choice modelling and the use of neural networks to try and solve the efficient runway allocation problem. These works have shown that weather, traffic demand and noise abatement are all important to the ATC decision process. The current study contributes to the literature in several ways. One is that attention is directed to the use of the linear kernel Support Vector Machine and Logistic Regression classification algorithms to solve runway allocation problems. Another one is that it examines features that have not been previously investigated as predictors for runway allocation, namely type of airplane, airplane weight, and maintenance planning. Maintenance features, in particular, prove to be important when it comes to predicting runway allocation.

The two algorithms yield similar classification performance, with the Logistic Regression classifier performing slightly better. They achieve considerably better results than the majority baseline classifier and this yields a promising perspective for future research.

1 Introduction

On January 4, 2016, Schiphol Group published a news report noting that they expected a record number of visitors in the upcoming year. During the year of their 100th anniversary, Amsterdam Airport Schiphol (AAS) would welcome a total number of 60 million travellers. This forecast is in line with current academic research, stating a global growth in air transportation over the last 30 years (Busquets et al., 2015). While there are a lot of opportunities for economic gain in the rapid growth of air transportation, there is great competition among the European hubs and the growth also entails negative consequences (such as flight delays, increased risk of collisions, reduced air quality, noise pollution and climate change) (Busquets et al., 2015; Kumar et al., 2008). As air travel continues to grow with approximately 5%per annum and expanding the number of runways is often impractical, it has become increasingly important for Air Traffic Control (ATC) to allocate runways more efficiently while maintaining safety and keeping in check with environmental regulations (Bennell et al., 2011). Therefore, it seems necessary to introduce a real-time decision aid to improve ATC operations.

1.1 Context

Controlling air traffic is a complex, highly collaborative task that requires swift responses to ever-changing situations (Mackay et al., 1998). In the process of assigning a runway to a flight movement (runway allocation), air traffic controllers have to consider many factors (such as the weather, traffic demand and environmental considerations) to make their final decision. This means that the capacity and safety of an airport are dependent on human decision-makers that experience a great amount of workload (Ramanujam and Balakrishnan, 2015). The current study tries to reduce air traffic controllers' workload and enhance ATC operations by creating a realtime decision aid for allocating runways.

There has been a considerable amount of research on modelling the ATC decision process (see Section 2). It is well known that weather (primarily cloud ceiling, visibility, wind speed and wind direction), traffic demand and noise abatement at an airport are major factors in runway allocation. Thus,

AAS, with its unstable weather conditions and complex noise preferential runway system, is especially interesting to evaluate a decision aid for ATC (Hesselink and Nibourg, 2011). Hesselink and Nibourg (2011) add that the availability of runways and Instrumental Landing Systems (ILS)¹ are equally important factors.

Taking into account the major factors in runway allocation mentioned in the literature, the present study uses supervised multi-class classification algorithms to predict the runway an air traffic controller is likely to allocate to under certain conditions. By combining data from various sources (such as weather data, traffic data and airplane data), a dataset of around 3.5 million flights at AAS between January 2008 and March 2016 was created (see Section 3). This dataset is used to predict the target feature **Runway**.

1.1.1 Academic Relevance

Although several studies have been conducted on modelling runway allocation (Avery and Balakrishnan, 2015; Bertsimas et al., 2011; Gilbo, 1993; Heblij and Wijnen, 2008; Janic, 2007; Kuiper et al., 2011; Nogami et al., 1996; Ramanujam and Balakrishnan, 2011, 2015; Zhang and Kincaid, 2014) (for a detailed overview see Section 2), little attention has been paid to the Stochastic Gradient Descent (SGD) approach to discriminative learning of linear classifiers to solve this issue. Most of the existing research is done with the use of econometric models and optimisation (Bertsimas et al., 2011; Gilbo, 1993; Heblij and Wijnen, 2008; Kuiper et al., 2011; Zhang and Kincaid, 2014). Research has also focused on improving the existing techniques by applying alternative modelling methods, such as discrete-choice modelling using utility functions (Avery and Balakrishnan, 2015; Ramanujam and Balakrishnan, 2011, 2015).

One of the ways in which a SGD approach is expected to improve forecasts on runway allocation is its efficiency and ability to handle large-scale machine learning problems. The ability to process an extended range of features and training examples is likely to improve the ability to capture the underlying behaviour of ATC. In fact, ATC is an especially interesting application for the SGD approach because it is characterised by a large and complex system that involves mixtures of (unstructured) data in various formats.

Because of the complexity of ATC, it is impossible to find an optimal solution for runway allocation and difficult to find a suboptimal one using only the known ATC heuristics (Nogami et al., 1996). A solution might minimise delays but cause an unnecessary amount of noise pollution to the environment. Fortunately, machine learning classifiers are widely known as

¹ILS enable aircrafts to land if pilots are not able to get a clear visual of the runway. They do this by using transmitted radio signals.

an effective method to extract new knowledge from data and generalise effectively. While the focus of econometric models is on optimising allocation (for minimising delays, noise or costs), this study tries to find a suboptimal solution within a practical computational time. Classification algorithms analyse the relationships between historical situations and their consequent best allocation decisions. The allocation decisions are defined by the current ATC who keep in mind the multitude of interests involved. These relationships are then used to predict future best practices, providing a suboptimal solution for the runway allocation problem. The SGD approach gives us the opportunity to do this quickly in real-time.

Another way in which this study contributes is the actual use of a large amount of data. The dataset used for this research consists of all 3.5 million flights at AAS from over the course of eight years (January 2008 to March 2016). By comparison, Ramanujam and Balakrishnan (2015) use two years' worth of data and Hesselink and Nibourg (2011) have done their research with just a single year of historical data.

1.1.2 Practical Relevance

For ATC, this study could be helpful in several ways. The first was already briefly discussed in this section and concerns the possibility to reduce workload for air traffic controllers. Research by Metzger and Parasuraman (2005) on conflict detection showed that automated decision aids could reduce mental workload and improve controller performance when functioning reliably. Reduced workload can increase efficiency and overall airport capacity (Gilbo, 1993). It is important to note, however, that reliability is a key factor in this. When the information was imperfect, automated detection performed worse than manual detection. Not only will flawed tools hurt capacity and efficiency of current ATC operations, they can also be of risk to airport safety (Metzger and Parasuraman, 2001). Since a predictive model can only be as good as its information, dependable information should be a top priority (Cole et al., 2000).

Besides reducing workload, runway allocation predictions can also be used to build new predictive models. One example is that good runway allocation predictions can lead to good noise predictions. Communities living around airports benefit from such a system as they will get insight into the traffic and amount of noise they can expect (Hesselink and Nibourg, 2011). Being informed is the first step in understanding and will reduce the number of noise complaints. Another example is the ability to build predictive maintenance models. This can serve construction companies to improve their activity planning by gaining insight in predicted runway use. For example, the company might decide to shift its planned maintenance to another day because the predicted runway use for that day is low. In addition, runway allocation predictions might be useful for maintenance contractors who can anticipate on asset usage and who can, in turn, predict their maintenance accordingly. A third example is capacity prediction. The choice of runways is an important factor in determining airport capacity. Airport capacity predictions are a key input to a range of air traffic management functions, such as air traffic flow management and airport surface operations scheduling (Gilbo, 1993; Ramanujam and Balakrishnan, 2015). Other implementations where runway allocation predictions could be useful are arrival management and operations scheduling by airlines (Hesselink and Nibourg, 2011). Hence, it proves to be very useful to predict runway allocation.

1.2 Research Questions

The aim of this research is to predict runway allocations for individual flights at AAS. I therefore formulate my research question as follows:

- Which features influence runway allocation?
- To what extent do the existing features influence runway allocation and which features are most influential?
- Based on the existing features, can predictions be made about runway allocation?
- Do these predictions yield better results than the majority baseline classifier?

To answer the first question, the existing literature is reviewed in Section 2. The following features were identified as relevant in runway allocation: weather (primarily cloud ceiling, visibility, wind speed and wind direction), traffic demand, noise abatement and maintenance work.

The second question will be dealt with in Section 3. An ablation study is performed to examine the effects of individual features on the performance of the model. As none of the features caused a large difference in Classification Accuracy (CA) after being omitted, it was decided to try and omit feature subsets. In this new experiment, the subset with weather features deteriorated CA most severely and can thus be seen as most influential.

The answer to the third research question is reported in Section 4. Here, the performance measures CA, Precision, Recall and F_1 -scores on the test set are reported and evaluated. The highest CA score (0.56) was achieved by the Logistic Regression classifier. The linear kernel Support Vector Machine (SVM) has a similar CA performance of 0.55. For the final research question, the performance scores of the linear kernel SVM and Logistic Regression classifiers are compared to the performance of the majority baseline classifier in Section 4. As the majority baseline was set at 0.21, both classification algorithms perform well beyond the baseline.

1.3 Structure

The remaining structure of this thesis is as follows:

Section 2 places the current research in a broader context and discusses our theoretical framework. Section 2.1 reviews previous academic work and Section 2.2 highlights the research gaps that were identified in the literature. Section 2.3 elaborates on the current study and justifies its added value.

Section 3 describes the experimental setup of this study in detail. Section 3.1 holds the description of the dataset. Section 3.2 describes the cleaning and pre-processing of the raw dataset and the subsequent experimental procedure can be found in Section 3.3. Finally, Section 3.4 reports which evaluation scheme and which error measures were used as evaluation criteria. This section provides all the necessary information to replicate the current study.

Section 4 reports the results that were found from the experiments. It reports the final CA scores on the test set. In Section 4.1, the classifiers' performance is discussed in relation to each other, to the performance on the validation set and to the literature. In Section 4.2, the majority base-line classifier is introduced and compared to the average CA scores of the classifiers. The performance measures Precision, Recall and F_1 -scores are discussed in Section 4.3.

Section 5 evaluates the results with regard to the research questions. Section 5.1 discusses the limitations of the current study. Its contribution to the existing framework is found in Section 5.2 and suggestions for future research are noted in Section 5.3. Finally, the implications for the field are discussed in Section 5.4.

2 Related Work

In the past three decades, global air transport has grown substantially. This rapid industry growth has led to airport congestion and significant flight delays at the busiest airports around the world. Moreover, it has been estimated that over the next 15 years air traffic in Europe and the US will more than double, perhaps even triple (Bennell et al., 2011; Busquets et al., 2015). Thus, many airports feel the need to increase their capacity and maximise throughput. But as airport capacity expansion projects tend to be expensive and take several years to complete, more efficient use of existing airport capacity (by means of improved planning and scheduling) can offer a less expensive and important part of the solution (Avery and Balakrishnan, 2015). One area where there is room for capacity expansion by improved planning is the runway system. Since the runway system is one of the most critical operational bottlenecks for the lion's share of airports, efficient runway utilization is a crucial instrument for capacity management and an especially important topic.

At any time, ATC chooses a selection of runways (and their associated traffic directions) to handle current flights. The subset of runways — known as the runway configuration — is of great influence on airport capacity (Gilbo, 1993). At AAS, a runway configuration can consist of up to 4 runways. The remaining runways are not used until there has been a switch in runway configuration. During arrival surges, a runway configuration typically consists of two arrival runways and one departure runway. During departure surges, ATC assigns two departure runways and one arrival runway. This process is a necessity to ensure safety and to meet aircraft separation requirements. Once a runway configuration has been decided upon, ATC assigns individual runways out of this selection for arriving and departing flights.

To perform the consequential task of allocating flights to runways, air traffic controllers have to consider many factors such as the weather, predicted traffic demand and environmental regulations, most important noise abatement (Ramanujam and Balakrishnan, 2015). On top of that, there may be a number of conflicting interests concerning the allocation of flights to individual runways. More often than not, different runways are preferred for different reasons. For example: in a particular situation, one of the runways is preferred for noise reasons, whereas another one is preferred for safety to the surrounding environment. This problem is illustrated by an accident that occurred at AAS in 1997, when an arriving aircraft was blown off the runway by a strong crosswind. After the crash, investigation showed that the aircraft was assigned to the runway because of noise considerations, while another would have been more favourable considering the wind conditions at the time (Heblij and Wijnen, 2008).

At AAS, ATC personnel handles approximately 1200 flights a day. Evidently, the high complexity and multitude of factors involved in this allocation task leads to a great amount of workload (Mackay et al., 1998). To assist in the efficient use of airport capacity and minimise air traffic controller workload, many models have been developed to function as real-time decision aids in runway allocation.

2.1 Previous Work

The last three decades have seen the emergence of a growing body of literature on runway allocation. In this section, the existing literature is reviewed by highlighting some of the most important studies in the field. The sections are organised according to the main methodology used in the study.

2.1.1 Optimisation

By far the largest amount of research on runway allocation has made use of optimisation techniques (Bennell et al., 2011; Bertsimas et al., 2011; Gilbo, 1993; Heblij and Wijnen, 2008; Kuiper et al., 2011; van Leeuwen et al., 2002; Stojković et al., 2002; Zhang and Kincaid, 2014). Optimisation techniques try to find the best alternative (in terms of cost or performance) under the given constraints, by maximizing favourable outcomes and minimizing unfavourable ones. While most optimisation research has focused on optimising the choice and sequence of runway configurations, Heblij and Wijnen (2008) concentrate — like the current study — on individual runway allocation. Their work describes the development of a runway allocation optimisation model that optimises with respect to delay, noise and safety. Additionally, operational runway usage, wind conditions and runway capacity are also taken into account. The multi-objectiveness of their optimisation model is unique in that the majority of studies aim only to minimise delay and its appropriate costs. It allows for the possibility to perform a trade-off between multiple objectives, providing ATC with a more practical solution.

To conduct their research, Heblij and Wijnen (2008) created three objective functions for their mixed integer linear programming model: one

for delay, one for third-party risk¹, and one for noise. The delay objective function is a summation of all delays occurring in all periods, as a result of the choices for certain runway configurations. Third party risk has a similar objective function, where the total risk for each flight movement is calculated by multiplying risk at a certain location with the number of inhabitants. Third party risk is the summation of the total risks of individual flight movements. Finally, the noise annoyance objective function is a cost function that penalises both high noise levels and a high number of inhabitants within the footprint of the flight. For every flight movement, the number of people living within a noise levels of ≥ 60 dB(A) is summed. The summation is the total noise annoyance.

These three objective functions are then combined in a new objective function to reach a final solution. This procedure is based on a weighted sum method. Instead of being determined beforehand, the weights are determined automatically such that all three objectives become equally important when they reach their absolute minimum. The model can now generate a new solution that will minimise for all three objectives simultaneously. It starts from a minimum delay solution, where delay is allowed to increase a few seconds in order to improve the other two objectives. When comparing the performance of the optimised results with a reference scenario of runway usage by airport authority, both noise annoyance and third-party risk dropped by almost 30%.

Optimisation techniques have some inherent limitations when it comes to implementing them for complex problems like ATC. In ATC, it is impossible to find an optimal solution, as it deals with a large number of conflicting interests (Heblij and Wijnen, 2008; Nogami et al., 1996). Therefore, optimisation models need to make assumptions about certain preferences of objectives incorporated in the model. In addition, large optimisation models have a significant computational time and are therefore limited in their ability to process large amounts of data (Bertsimas et al., 2011; Kuiper et al., 2011). The SGD approach to discriminative learning of classification algorithms — as used in this study — tries to find suboptimal solutions within a practical computational time. This kind of solution can be very useful to ATC.

2.1.2 Heuristics

An alternative to optimisation models based on linear and integer programming can be found in the paper by Janic (2007). In his work, a heuristic algorithm is developed for runway capacity allocation, to minimise (current)

¹The probability of a citizen living in the surroundings of an airport losing his or her life due to an aircraft collision in a given year.

delay costs. Like classification algorithms, heuristic algorithms find a solution close to the best one within a short time-frame. Unlike optimisation models, the heuristic algorithm is based on 'greedy' criteria that closely reflect the rules of thumb used by air traffic controllers. Using greedy criteria implies choosing the best option for the current moment, without considering possible future events. In this case, one can only hope that making the best choice today will lead to an optimal solution for the future.

As a reason for his work, Janic (2007) argues that the inherent complexity of optimisation models might make it difficult to understand and implement them, and therefore, decrease the likelihood of air traffic controllers embracing the method. His approach might offer a more accessible solution, as it is more closely related to the still preferred rules of thumb, while still obtaining decent results.

While the heuristic algorithm is computationally less expensive than most optimisation models, it also achieves worse result. In all experiments, the heuristic algorithm performs weaker than the benchmarking (optimisation) models. The differences in total output range from a decrease of 1.4% to 2.4% in all cases. As the algorithm is built to closely represent the rules of thumb used by air traffic controllers, this suggests that there is room for improvement of the current ATC operations. Still, Janic (2007) contends that the algorithm appears to qualify since it might be very difficult to notice such small differences.

2.1.3 Discrete-choice Models

A smaller, recently developed field of research is the use of discrete-choice models and utility functions in runway allocation (Avery and Balakrishnan, 2015; Ramanujam and Balakrishnan, 2011, 2015). Although discrete-choice models have been widely used in fields like transportation modelling, the use of this approach in ATC is relatively new. All studies in this field focus on runway configurations. However, they still have their implications on individual runway allocation, due to their similar decision process. A key novelty in the work of Avery and Balakrishnan (2015) is the understanding of the ATC decision process. In short, discrete-choice models are behavioural models in which decision makers have to choose among an exhaustive set of mutually exclusive alternatives, called a choice set. The utility functions used in the models are defined as linear functions of attributes that can influence the decision. The importance of the individual features is captured by the assigned weights in the utility functions. It is assumed that the suitable alternative with the maximum utility will be selected by the decision maker. Because of this, the estimated utility functions and the discretechoice model can be used to calculate the probability of choosing a runway configuration alternative, given the values of their corresponding features. The runway configuration with the maximum probability of being chosen is selected as the predicted runway configuration.

The utility functions of the model by Avery and Balakrishnan (2015) includes features such as wind speed and direction, arrival and departure demand, cloud ceiling, visibility, coordination with neighbouring airports and noise abatement procedures. While most research has tried to quantify noise, Avery and Balakrishnan (2015) use a categorical version, making a distinction between flights that arrive and depart over the suburbs and flights that arrive and depart over water (San Francisco Bay). As the aim of their research is to predict runway configurations, features like inertia (the preference of air traffic controllers to stay in the same configuration) and switch proximity (resistance to configuration changes) are also included. Avery and Balakrishnan (2015) state that their discrete-choice model results in a prediction accuracy between 82% and 85%, depending on the quality of the data available. The model is also tested on weather forecast data and scheduled demand, as this generates a better idea of how the model would perform in real situations, where there is only prospect data available. This model generates an accuracy score between 79% and 82%.

The prediction approach is contradictory to both optimisation models and the heuristic algorithm, as it does not try to optimise anything, but rather learns from the nominal behaviour of air traffic controllers and tries to predict their future decision behaviour. This can cause limitations as the model does not account for potential variability in the quality of decisions. Different decision-makers, who have varying experience, preferences and rationales will decide differently when put in the same situation. This bias is a problem for the current research as well, as it is assumed that past decisions of ATC are best practice.

2.1.4 Neural Networks

Nogami et al. (1996) use neural networks to create a real-time decision support for ATC. According to Nogami et al. (1996), ATC is operating in a dynamic environment, full of stochastic elements and processes that are difficult to formulate mathematically. Other characteristics of ATC defined by Nogami et al. (1996) are the conflicting interests and the great number of different control variables and constraints involved. They explain that it is impossible to find an optimal solution for the ATC problem without interrupting operations as this is too computationally heavy. Techniques like optimisation use exhaustive search-based methods, whereas for on-line decisions it is more useful to find a suboptimal schedule to be made in real time.

Neural networks are widely known as powerful methods to extract new knowledge from (unstructured) data and generalise effectively. In the work of Nogami et al. (1996), inputs used for the model to approximate current traffic situations are nominal flight plans, meteorological forecasts, radar data, and restrictions imposed by flow control. Among the possible traffic situations, possible anomalies like aircraft accidents are also included. The output of the model is the best heuristic rule used in that traffic situation. Nogami et al. (1996) use the inductive learning backpropagation algorithm to assign weights to the connections between nodes of the neural network layers and in this way, acquire knowledge about the relations between traffic situations and best control actions. Depending on the amount of data the neural networks were given, prediction accuracy could rise up to 90%.

As with discrete-choice models, neural networks try to learn the relationships between past situations and the behaviour of air traffic controllers. Again, this approach does not account for variability in the quality of decision and can introduce bias. Also, in their limitations section, Nogami et al. (1996) argue that neural networks can sometimes be bad at extrapolation. However, as the aim of this research is to find a sub-optimal but satisfactory solution within a short time span, these issues can be considered trivial.

2.2 Research Gaps

To the best of my knowledge, little attention has been paid in the literature to the use of the SGD approach to discriminative learning of linear classifiers to solve the runway allocation problem. In the current study, the linear kernel SVM and Logistic Regression classification algorithms are used. Like discrete-choice models and neural networks, these classifiers are useful when we want to learn the mapping $\mathcal{X} \mapsto \mathcal{Y}$, where $x \in \mathcal{X}$ is some object and $y \in \mathcal{Y}$ is the class label. They estimate the mapping relations between the given input and output sets \mathcal{X}, \mathcal{Y} and try to find a suitable $y \in \mathcal{Y}$, given a previously seen $x \in \mathcal{X}$. While SVM is based on the simple idea of finding the hyperplane with the largest margin to separate data classes, it often leads to high performances as it still constructs models that are complex enough to handle real-world applications (Hearst, 1998). Logistic Regression is similar to the linear kernel SVM, as it also tries to linearly separate the (hyper)space of features. The only difference between them is that Logistic Regression does not try to maximise the margin between the classes, but instead tries to minimise the error of the predicted values to the observed values.

While most research has focused on optimising the choice of runway configurations, the current study concentrates on individual runway allocation. Predicting individual runway allocation contributes to the current body of literature as in some situations, where configurations consist of multiple runways, this approach can be very useful. For example: in the situation where a configuration consist of one departure runway and two arrival runways, an air traffic controller still has to decide which runway is most optimal for an approaching flight. A decision-aid that is specified for individual runways can assist in that process. It can also be used as an extra check of whether the current runway configuration is most optimal. Additionally, Heblij and Wijnen (2008) explain that predicting individual runways for AAS is not a problem as the airport does not have many highly dependent configurations². As configurations remain mostly independent, it is possible for the model to assign freely. However, for other airports that are using highly dependent configurations more regularly, it is preferred that the model will be extended with runway configurations.

2.3 Current Study

Since little attention has been paid to the linear kernel SVM and Logistic Regression classifiers — that are deemed to be very useful — this study will focus on their use. Both the linear kernel SVM and Logistic Regression are linear supervised algorithms. In this research, they will be used to solve a multi-class classification problem. SVM can be used for both classification and regression. Each data item is plotted as a point in a *n*-dimensional space, where *n* is the number of features. In classification, it tries to find the hyperplane that provides the maximum separation between the different classes (Vapnik, 1995). Logistic Regression is a classifier that can solve binary problems. For multi-class classification problems like the current one, it uses an *one-vs-all* approach. It creates an S-shaped curve with the probability estimate of labels, which is very similar to the step wise function. The classifier tries to find the best-fit logistic function for the data and then implements a threshold of 0.5 to divide the data into two classes.

Given a set of training examples $(x_1, y_1), \ldots, (x_n, y_n)$, the classifiers learn a linear scoring function. For both classifiers, the linear model formula for the dependence of the predicted target on the features can be expressed by:

$$z = \mathbf{w} \cdot \mathbf{x} + b, \tag{2.1}$$

where z is the score of the linear model, x is the feature value, w is the feature weight and b is the intercept. In order to make predictions, we simply look at the sign of z for the SVM with linear kernel.

 $^{^{2}}$ As runways intersect or are situated parallel to each other with a distance closer than 762 meters (or 2500 feet) they can become dependent, meaning that the separate runways cannot always be used at the same time. Jet blasts (rapid air movement produced by the engines of an aircraft, particularly during take-off) and the use of a single runway for both arrivals and departures (the so called mixed mode) can also cause dependency.

A Logistic Regression model uses the inverse logit function to predict:

$$(p)_{\text{pred}} = \text{logit}^{-1} \cdot z, \qquad (2.2)$$

where $(p)_{\text{pred}}$ is a predicted probability score, varying between 0 and 1. SVM tries to maximise the margin between the classes. The *hinge loss* function that the soft margin SVM tries to maximise is:

$$\ell_{\text{hinge}}(z) = \max(0, 1 - y \cdot z), \qquad (2.3)$$

where y is the observed value of the class, either 0 or 1. The log loss function (or cross-entropy), which quantifies the mistakes of the Logistic Regression classifier, can be denoted as follows:

$$\ell_{\log}(z) = -y \log(p_{\text{pred}}) - (1 - y) \log(1 - p_{\text{pred}}).$$
(2.4)

For each instance, only the predicted class actually contributes to the sum. The Logistic Regression classifier tries to minimise the log loss function to find the model which gives the maximum probability to the training targets.

Hence, the present work uses a SVM with linear kernel and a Logistic Regression classifier to analyse a dataset that consists of flights at AAS from January 2008 until March 2016. The goal of this research is to create a decision-aid that helps air traffic controllers in making their allocation decisions. Using these two classifiers, the relationships between historical situations and best allocation decisions is learned and applied to future flight movements.

By talking to professionals from the field, it became clear that primarily weather (especially wind direction), but also noise abatement regulations and runway maintenance were major factors in the allocation process. Academic literature such as work the by Avery and Balakrishnan (2015); Ramanujam and Balakrishnan (2015) also showed that meteorological conditions like wind speed, wind direction, height of the cloud ceiling and visibility, traffic demand, noise abatement, inter-airport coordination (only in dense areas with multiple airports that are close together, like New York City), inertia (the aversion to changing runway configurations) and configuration proximity (the amount of workload associated with a certain runway configuration switch) were important. As the last two factors are associated with runway configurations specifically and inter-airport coordination is not directly applicable for AAS, only traffic demand was seen as a relevant addition to this research. This answers the first research question (Which features influence runway allocation?).

In order to really recreate the ATC allocation decision process, as many as possible decision factors were included in the dataset. Unfortunately, not all important features could be added due to a lack of access to this data. For example, it was not possible to add precise runway maintenance data to the dataset as this data was not available. However, it was possible to incorporate maintenance planning in the dataset. A large number of weather features and several features about traffic demand were also included. As AAS has different runways of different sizes that cannot host all types of airplanes, airplane type and weight are expected to be important predictors as well. These features, along with the maintenance features, are a key novelty of this research. A full description of the dataset can be found in ??.

With this dataset, we can answer the remaining research questions. The answer to the first research question (Which features influence runway al*location?*) was solely based on academic literature and domain knowledge from professionals in the field. Weather (especially wind direction), noise abatement regulations, runway maintenance and traffic demand are major factors in the decision making process of ATC. The second question (To what extent do the existing features influence runway allocation and which *features are most influential?*) is answered in Section 3, where the results are reported of omitting each feature from the dataset once. The third research question (Based on the existing features, can we make predictions about runway allocation?) is answered in Section 4, where the performance measures CA, Precision, Recall and F_1 -scores of the model are reported. And finally, the answer to the fourth research question (Do these predictions yield better results than the majority baseline classifier?) is reported in Section 4 as well, where the CA scores of the linear kernel SVM and Logistic Regression models are compared to CA score of the majority baseline classifier.

3 Experimental Setup

Three data sources were used to create the dataset used in this research. The first data source was obtained by contacting AAS directly. As this data is confidential, it will not be made publicly available. However, it will be described and the code that was used to manipulate it will also be made available. The data from AAS holds the number of similar flights per day (between 2008 and March 2016). Flights are grouped by date, airplane type, maximum take-off weight, arrival or departure and the runway that was used. For each group of similar flights, the amount of times that they occurred is stated. The data was delivered in eight separate documents, one for each year. In total, the files contain 622,029 instances and 7 features.

The second dataset was retrieved to obtain data about the weather conditions at AAS. It was obtained online from the Royal Netherlands Meteorological Institute (KNMI). KNMI collects data at its weather station at AAS and makes it publicly available. The dataset contains 23,802 instances (one instance for each day, starting from 1951) and 41 features, all related to the weather (e.g. temperature, visibility and sunshine duration).

The third and last data source was the maintenance planning for the years 2010 to 2016, retrieved from the website of Bezoekers Aanspreekpunt Schiphol (BAS). These were images of calenders that contained the planned maintenance for each runway at AAS for one year. Because maintenance sometimes is delayed due to weather circumstances, the data from this source cannot be trusted completely. However, as actual maintenance data has not been collected by BAS, this is the most viable option available.

Any code and (part of) the data needed to reproduce this research can be found in my Github repository¹.

3.1 Dataset

The final dataset used in this study contains 3,529,268 instances and 48 features. It contains features about the date, type of airplanes, weather

¹http://github.com/kiaeisinga/thesis

circumstances, maintenance planning and past traffic demand. The target attribute of the dataset, Runway, is a discrete variable with 12 classes. These classes are runway directions of six different runways. Each runway has two separate runway directions, because there are two directions in which planes can use the runway. For example, the Zwanenburgbaan at AAS is represented by the directions 18C and 36C. The runway directions represent the vector degrees (360=north, 90=east, 180=south, 270=west) on which the runway is situated, divided by ten. The letters (R=right, C=centre, L=left) are used to specify which runway is meant when they have the same value for vector degrees, as there are three runways that run parallel to each other. To further clarify this, Figure 3.1 shows the current runway map of AAS. All 48 features included in the dataset are presented in a detailed overview in ??.



Figure 3.1: Runway map of Amsterdam Airport Schiphol

3.1.1 Exploratory Data Analysis

Exploratory Data Analysis (EDA) can provide a quick grasp on the dataset and summarize its main characteristics. The following graphs help create a deeper understanding of the dataset.

1. There is an overall increase in the amount of air transportation at AAS.



Figure 3.2: Air traffic growth at Amsterdam Airport Schiphol

Although this plot is not directly related to the current research question, it proves the importance of this research. The graph shows that there is an overall increase in air transportation, a phenomenon that was previously mentioned in the literature. Simultaneously, the literature suggested that there was an increased importance to increase efficiency of ATC operations because of this, which is the goal of this research.

 The majority of the flights are allocated to the Kaagbaan (06/24) or the Polderbaan (18R/36L). The Oostbaan (04/22) handles the least amount of flights.



Figure 3.3: Total number of flights for each runway

The reason for this is that the Kaagbaan (06/24) and the Polderbaan (18R/36L) have extended areas that are lowest in habitation. Using these runways produces the least amount of annoyance in surrounding areas and is therefore beneficial. In fact, at night, these two runways are used exclusively. In addition, the Polderbaan (18R/36L) is the longest and broadest runway available at AAS and can thus handle a wide variety of airplanes. On the other hand, the Oostbaan (04/22) is the shortest and oldest runway at AAS. It usually is not considered as a functional option by ATC. Nevertheless, it is used to handle lightweight ($\leq 7,000$ kg) airplanes. As shown in Figure 3.4, the Oostbaan (04/22) handles the greatest proportion of lightweight airplanes.

Figure 3.3 also shows that there is imbalance in the dataset — meaning that not all classes are equally represented. This might cause a difference in predictability per-class later on.





Figure 3.4: Total number of light airplane flights ($\leq 7,000$ kg) for each runway

4. Most arrivals are allocated to runway 24 (Kaagbaan) and most departures are allocated to runway 18R (Polderbaan).



Figure 3.5: Arrivals and departures on each runway

At AAS, each runway can be used in two directions. As airplanes always need a headwind to arrive at and depart from airports, the availability of runway directions is highly dependent on the wind direction at the moment. In practice, as shown in Figure 3.5, this means that there is an arrival and departure direction at each runway. Even though the ratio between arrivals and departures at AAS is equal, an individual runway direction is primarily used for either arrivals or departure.

5. In the case of a northern wind, 06 (Kaagbaan) and 36L (Polderbaan) are most preferred, whereas with a southern wind, 24 (Kaagbaan) and 18R (Polderbaan) are most preferred.



Figure 3.6: Flight distribution by wind direction

Runway preference varies by wind direction. In the Netherlands the most common wind direction is south-west. In case of a south-western wind, the runways 24 (Kaagbaan) and 18R (Polderbaan) are most often used. However, in the case of northern wind, 06 (Kaagbaan) and 36L (Polderbaan) are most preferred. This supports the claim from the previous finding that wind direction has a great impact on runway allocation.

3.2 Cleaning and Pre-processing

After collecting data from AAS, KNMI and BAS, the first step was to preprocess all the obtained data in R. The AAS data was stretched so that each instance would now reflect one flight. This way, the dataset can later on predict the runway allocation for each flight with all the involved influence factors. As the AAS dataset starts from January 1, 2008, the weather data prior to this date were removed from the KNMI dataset. As the BAS data consisted of images of calenders, the dates were transcribed from the images and loaded into R using an ifelse-statement. Six maintenance features — one for each runway — were created, where a value of 1 was given to all dates when maintenance was planned.

3.2.1 Missing Values and Feature Extraction

Having missing values in the dataset can reduce the ability of a model to classify correctly, as the estimates may be biased. Therefore, it is important to handle missing values carefully.

The missing values from the AAS dataset were removed, as they only made up 0.0063% of the dataset. The KNMI dataset contained three missing values for a single date, that were also deleted as they only made up 0.0335% of the dataset. The BAS data contained missing values for all dates in 2008 and 2009, as no maintenance planning was made for these years. These cases were all imputed with the value 0(=No Maintenance), as this value is true for 97.16% of the cases in the dataset.

As the features Last.year and Yesterday represent the total amount of flights from the same date last year or from the day before, naturally, some missing values were created. For all dates in 2008 (443,163 instances or 12.5568% of the dataset), no values were available for the Last.year. The same holds for the feature Yesterday on January 1st, 2008 (776 instances or 0.0220% of the dataset). In order to deal with these missing values, the mean was imputed for both features.

In addition, all features in the KNMI dataset that were described as 'Hourly division in which ... was measured' were deleted from the dataset, as these features do not contain information about the weather. The remaining features were kept in the merged dataset as this enabled for an ablation study later on to see which features are most important for prediction.

3.2.2 Outliers

By examining the summary statistics of the dataset, no obvious outliers were found in the dataset. However, once a new feature called Total.flights was created for EDA, it became evident that there were some outliers here.

Whereas the mean Total.flights — which gives the total number of flights on a particular day — was 1461, its minimum was 2. As the first quartile of this feature had a value of 1111, this seemed very odd. An illustration of the data distribution of the feature can be found in Figure 3.7.



Figure 3.7: Data distribution of Total.flights feature

From this violin plot, it is obvious that there are some outliers in the dataset. When checking for the date, April 17, 2010, it became clear that the cause of this low number of flights is not due to poor measurement, but rather due to a volcano eruption in Iceland, spreading ashes and causing AAS' airspace to be unfit to fly in. The disturbance lasted several days, from April 15, 2010 to April 20, 2010. Also, on May 17, 2010, a similar event happened where a new ash cloud disrupted the daily operations.

When creating a subset for all instances that had values less than 801 $(\mu - 3\sigma)$ for Total.flights, it became clear that more unusual events had had their influence on air traffic at AAS. While most of them have to do with bad weather (heavy snowfall or strong wind), some are more unusual events like a failed terrorist attack on December 25, 2009 and labour strikes from the air traffic controllers in Spain on December 4, 2010. Also, on the first day of Christmas (December 25) of each year and sometimes also on New Year's (December 31 and January 1), the number of flights is relatively low.

However, as this is not faulty data and it was not clear whether these outliers actually changed runway allocation, it was decided not to instantly remove these dates. Rather, an experiment was run to see which dataset (with or without these outliers) created the best classification results. As removing the outliers (either just the ones from the volcano eruption in Iceland or the one below the arbitrary boundary of $\mu - 3\sigma$) did not improve the performance of the model, no outliers were removed from the dataset.

3.2.3 Feature Engineering

As mentioned in the previous section, a new feature called Total.flights was created for EDA. It helped identify the general trend of air traffic growth and to find some unexpected outliers. Another feature, called Name, was also created for EDA. It indicates which runway directions (e.g. 18L) belong to each runway (e.g. Aalsmeerbaan). However, these two features were deleted from the dataset after EDA, as they hold information that cannot be known about future flights and thus could not be used as predictors in the classification task.

Other features that were engineered were Weekday, Season, Weekend, Icao.wtc, Wind.discrete, Vmc.imc, Beaufort, Last.year and Yesterday. Weekday, Season and Weekend were all derived from the feature Date and can help recognise weekly or seasonal effects in runway allocation. Icao.wtc was created from the feature MTOW and divides airplane types into four weight classes (Light, Medium, Heavy and Super Heavy) according to the official guidelines by the International Civil Aviation Organisation (ICAO)². The feature Wind.discrete was generated from the feature Wind.direc, where each wind direction was categorised into one of four categories (North West, North East, South West and South East).

The feature Vmc.imc classifies visibility into two classes: Visual Meteorological Conditions (VMC) and Instrumental Meteorological Conditions (IMC). Because AAS is a so-called class A airspace, all planes fall under instrument flight rules. This means that all airplanes are separated by ATC to ensure safety and need ATC clearance to arrive and depart from AAS, even when visibility is sufficient. Visual flight rules, where air planes are allowed to separate autonomously under sufficient visibility conditions, are only used at AAS by high exception. At AAS, the VMC minimum (visibility \geq 5 km) in state for visual flight rules is used as information for pilots, but does not mean that visual flight rules are at force. However, this information can be used to create a division between VMC (visibility ≥ 5 km) and IMC (visibility < 5 km) in the dataset. As not all runways at AAS are equipped with ILS and thus cannot be used during low visibility weather conditions, this feature should help make the relationship between visibility and runway allocation become more evident. Two features were used to construct Vmc.imc: Visibility.min and Visibility.max. When Visibility.min was ≥ 5 km, Vmc.imc was categorised as VMC and when Visibility.max was < 5 km, it was categorised as IMC. It was possible to determine the

²http://www.skybrary.aero/index.php/ICAO_Wake_Turbulence_Category

conditions for 2,094,250 (out of 3,527,563) of the instances. The remaining cases were labelled as Unknown.

The Wind.speed.daily feature allowed for the creation of Beaufort, a wind speed categorisation which was created according to the official Beaufort scale categories³. Finally, the features Last.year and Yesterday were extracted from the feature Total.flights to see if the amount of traffic demand has any influence on runway allocation.

3.3 Experimental Procedure

Since the dataset contains 3,529,268 instances, it was crucial to find a classification algorithm that was able to work with large datasets. As soon as the pre-processing phase was finished, the dataset was loaded into Python. Here, the categorical features Date, Icao.type and Act.description were deleted from the dataset to save memory, as they contain lots of levels. The remaining categorical features (Weekday, Season, Arr.dep, Icao.wtc, Wind.discrete and Vmc.imc) were transformed into dummies. The dataset now consists of 61 columns.

Then, the dataset was split into a training, validation and test set. As the goal of this research is to start predicting for recent flights, the data was split by year, where the most recent years were used as a test set. This means that the years 2008 until 2012 were used for training, 2013 and 2014 were used for validation and 2015 and 2016 were used as a test set.

All the continuous features in the dataset operate on various ranges. When applying classifiers that operate on Euclidean distances it is necessary for the input to have equal distances. Therefore, the continuous features are standardised. Standardisation is applied to all the sets because the classifiers will not be able to work properly if the features of the different sets are not similar. The new features with standard scores are used to replace the old features in the training, validation and test set.

An on-line SGD approach to discriminative learning of linear classifiers from the scikit-learn library in Python was used to train the data, as it is able to handle large datasets very well. SGD is the stochastic approximation of the Batch Gradient Descent optimisation method. Like the Batch Gradient Descent, it learns weights that would minimise the squared loss of the function. But instead of going through the entire dataset each time like the Batch Gradient Descent, the SGD estimates the gradient of the loss each sample at a time. The model starts with w = 0 and updates the weights along the way. This way, it is able to go through the entire dataset much faster. In addition, it allows us to predict real-time, which is one of the aims

³http://projects.knmi.nl/hydra/faq/druk

of this research. SGD supports several loss functions: setting the parameter loss to *hinge* or *log* allows us to work with the linear kernel SVM and Logistic Regression classifiers, respectively.

3.3.1 Parameter Grid Search

A grid search was performed to find the most optimal parameters for the linear kernel SVM and the Logistic Regression classifiers. The parameters **n_iter** (number of iterations) and **alpha** (cost) were optimised. The number of iterations is equal to the number of passes over the training data. Increasing the number of iterations can help optimise the model and achieve the highest accuracy, but comes at the expense of computational time. For regularisation, the general penalising parameter cost sets the complexity of the model. It represents a trade-off between finding the hyperplane with the maximum margin between the classes and a hyperplane that correctly separates as many instances as possible. A large value of cost is better at correctly separating instances, but can cause complexity to the model and a danger of overfitting. A small value of cost prefers to find the maximum margin between the classes, sacrificing the fact that some instances will be misclassified. To find the most optimal values for both parameters, a nested for-loop was created to test all possible combinations of the parameters. The number of iterations parameter was tested for values 1 to 25, with steps of 1. The cost parameter was tested for values 0.00001, 0.0001, 0.001, 0.001, 0.01, 0.1, 1 and 10. Table 3.1 shows the results of this search: the highest CA found for each classifier with its corresponding best parameter values.

Algorithm	Cost	Number of iterations	CA
SVM with linear kernel	0.0001	19	0.54
Logistic Regression	0.0001	9	0.55

Table 3.1: Optimal parameters found by grid search

For both classifiers, 0.0001 is the value of cost that produces the highest CA. The number of iterations is different: the optimal values are 19 for the linear kernel SVM and 9 for the Logistic Regression classifier. From now on, these optimal parameters will be used to train the model on.

3.3.2 Feature Influence

To answer the second research question (To what extent do the existing features influence runway allocation and which features are most influential?), each feature was omitted once from the training and validation set to see its effect on CA. For the categorical features that were transformed into dummy variables, the set of columns representing a single categorical variable was treated as one unit and omitted all at once. The results of these experiments are presented in Chapter B. The features that decreased CA most severely when being omitted were labelled as most important to the prediction performance of the model. In general, wind direction (Wind.discrete), maintenance at the Buitenveldertbaan (Main.B) and daily temperature (Temp.daily) are the most important features according to the two classifiers used in this study, i.e., linear kernel SVM and Logistic Regression. While Wind.discrete was identified to be of great importance by previous literature, Main.B and Temp.daily are novel. As the current study is the first to introduce maintenance as a predictor for this problem, this defends its contribution to the existing research framework. For the Logistic Regression classifier, the feature for maintenance at the Oostbaan (Main.O) is also an important predictor. Surprisingly, Vmc.imc — a categorical feature for visibility — was a rather unimportant predictor for runway allocation, despite the fact that Heblij and Wijnen (2008) argue that ILS are of great importance to the availability and allocation of runways. Since all of the removals decreased the predictive performance of the model, all features remained in the dataset.

Noticeably, omitting separate features did not generate great differences in CA. Therefore, another ablation study was run where groups of features were omitted. Four groups of features were made according to topic: Weather, Maintenance, Airplane weights and Traffic. The Weather subset contains all features listed from Wind.direc until Vmc.imc in ??. The Maintenance subset consisted of features Main.A, Main.B, Main.K, Main.O, Main.P and Main.Z. Features MTOW and Icao.wtc created the Airplane weights subset and finally, the Traffic subset was made out of features Last.year and Yesterday.

The results of these experiments are presented in Table 3.2 for the linear kernel SVM classifier and in Table 3.3 for the Logistic Regression classifier.

Omitted feature subset	CA
Weather	0.50
Maintenance	0.52
Airplane weights	0.52
Traffic	0.53
None	0.54

Table 3.2: CA results with linear kernel SVM when omitting feature subsets

Omitted feature subset	\mathbf{CA}
Weather	0.51
Maintenance	0.53
Airplane weights	0.54
Traffic	0.55
None	0.55

Table 3.3: CA results with Logistic Regression when omitting feature subsets

As shown in Tables 3.2 and 3.3, the largest subset — Weather — is most predictive for both classifiers. It increases CA with approximately 4%. Maintenance, not used as a predictor in previous research, increases the CA of the model by approximately 2%. Airplane weights, also a novel predictor, contributes to the model with an increase of 1 to 2% in CA. The Traffic subset, while deemed to be an important predictor in previous literature, is the least predictive in the current experiments.

3.4 Evaluation Criteria

To evaluate the different models (in terms of parameters and feature subsets), a validation set was used. CA was the most important measure to evaluate the outcomes, as the goal of this research is to create an as accurate as possible decision aid for ATC. The parameter values that produced the highest value for CA were selected as optimal and used in the rest of the models. When comparing the performance of the model after omitting a feature or feature subset, (the decrease in) CA was also used as a criteria to signal the most important features.

Next, the insights of these experiments were used on the test set. The test set is preprocessed in the same way the train set is preprocessed and again, the same optimal parameters are used to train the model on. The results of this final experiment are presented in Section 4. In addition to CA, the performance measures Precision, Recall and F_1 -scores will also be reported.

4 Results

In this section, the performance results of the classification task are presented. It will provide an answer to the most important research question of this study (*Based on the existing features, can predictions be made about runway allocation?*). The current study focuses on introducing a real-time decision aid to improve ATC operations. It does this by using an on-line SGD approach to discriminative learning of linear classifiers to predict runways for future flights. The SGD is updated along the way and is therefore very efficient in processing large amounts of data and is able to predict in real-time. With SGD, it is possible to choose from a number of classifiers to train the model on, among them the linear kernel SVM and the Logistic Regression classifier used in this study.

As mentioned in Section 3, a grid search was used to find the optimal parameters for both algorithms. Section 3 also describes that individual features were tested to see their effect on the CA of the model. Since none of the removals improved the predictive performance of the model, all features were kept in the dataset.

Thus, the optimal parameters and all existing features were used to train the model and test the classifiers' performance on the test set. Table 4.1 shows the results of the models when used on the test set.

Algorithm	CA
SVM with linear kernel	0.55
Logistic Regression	0.56

Table 4.1: CA after testing on the test set

4.1 Performance of the Model

Both algorithms yield similar classification performance, with the Logistic Regression classifier performing slightly better. When comparing it to the results stated in Table 3.1, it is noticeable that the predictive performance on the test set is better than it is on the validation set. This could mean that the model was slightly underfitted. In that case, the model is too simple with regards to the data it is trying to model and it has a high bias. Increasing the cost parameter can increase the complexity of the model and ensure a better fit. Changing the parameters of the classifiers after evaluating on the test set is not advised, however, as it causes overfitting on the test set. The difference in predictive performance might also be caused by chance, when the test set is simply easier to classify than the validation set.

Compared to previous research (Avery and Balakrishnan, 2015; Nogami et al., 1996), which reports CA scores of 79% and more, these models perform considerably worse. However, this is a trivial point as the predictive performance of a model will always greatly depend on the available data: number of labels, balance between classes and available features and instances. The limitations of this study and its dataset will be further elaborated on in Section 5.

Hence, previous literature might not be a fair comparison. One way to justify the results of classifiers is by comparing them to the results of the baseline and showing that they are indeed better than majority predictions.

4.2 Baseline

To define the baseline for this study, a majority classifier was used. The majority baseline classifier is a classifier that predicts the most frequently used runway direction for each flight movement. Table 4.2 shows the frequencies of all class labels in the dataset and their corresponding percentages.

Runway	Count	Percentage	R	lunway	Count	Percentage
04	20988	0.59		22	70972	2.01
06	365919	10.37		24	723392	20.50
09	90561	2.57		27	204347	5.79
18C	276374	7.83		$36\mathrm{C}$	194377	5.51
18L	329360	9.33		36L	425032	12.04
18R	650959	18.44		36R	176987	5.01

Table 4.2: Number of flights per runway

As displayed by Table 4.2, runway direction 24 is most frequently used for flight movements at AAS. It is used in 20.5% of all cases. Therefore, a majority baseline classifier predicting 24 for each arriving or departing flight at AAS, will have a CA of 0.21. This is well below the achievements of the current classifiers. This means that we can answer our final research question (*Do these predictions yield better results than the majority baseline classifier?*) positively.

4.3 Precision, Recall and F_1 -score

Depending on the classification task, different performance measures are more important. As the goal of this study is to create an as acurate as possible decision aid for ATC, CA is deemed to be most important here. Therefore, all previous models were evaluated and selected to achieve the highest CA. In this section, other performance measures will also be examined, namely Precision, Recall and F_1 -scores. Precision tells us what proportion of flights that were allocated to (e.g.) runway 24 by our model, actually should have been allocated to 24. It can be denoted by the following formula:

$$Precision = \frac{TP}{TP + FP},\tag{4.1}$$

where TP is the number of True Positives (correctly identified as 24) and FP is the number of False Positives (incorrectly identified as 24). Recall is a little different: it tells us what proportion of flights that should have been allocated to 24, actually were allocated to 24 by our model. This can be expressed using:

$$Recall = \frac{TP}{TP + FN},\tag{4.2}$$

where TP is again the number of True Positives (correctly identified as 24) and FN is the number of False Negatives (incorrectly rejected as 24). Sometimes we want to have a single number to describe the performance of the model. Therefore, the two measures Precision and Recall can be used to compute the F_1 -score, which is their harmonic mean:

$$F_1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}.$$
(4.3)

The overall results of these performance measures for both classifiers are presented in Table 4.3.

Algorithm	CA	Precision	Recall	F_1 -score
SVM with linear kernel	0.55	0.49	0.55	0.47
Logistic Regression	0.56	0.51	0.56	0.49

Table 4.3: Average performance scores on test set

Again, both algorithms produce similar results, but the Logistic Regression classifier is performing slightly better on all performance measures.

4.3.1 Per-class Performance

To get a more detailed view on how the classifiers predict, we can have a look at their confusion matrices. The confusion matrices of the linear kernel SVM and the Logistic Regression classifier are presented in Chapter C. The diagonal numbers in the matrix represent the number of correctly identified classes. By dividing the correctly identified class by its row sum, we can compute its Recall. The same can be done for Precision by dividing the correctly identified class by its column sum.

The per-class results of Precision, Recall and F_1 -scores on the test set are presented in Table 4.4 for the linear kernel SVM classifier and in Table 4.5 for the Logistic Regression classifier.

Runway	Precision	Recall	F_1 -score
04	0.66	0.30	0.41
06	0.59	0.79	0.68
09	0.14	0.07	0.09
18C	0.39	0.03	0.05
18L	0.40	0.09	0.15
18R	0.57	0.91	0.70
22	0.58	0.21	0.31
24	0.60	0.79	0.68
27	0.41	0.20	0.27
36C	0.07	0.00	0.01
36L	0.54	0.88	0.67
36R	0.27	0.08	0.12

Table 4.4: Precision, Recall and F_1 -scores for linear kernel SVM

For the linear kernel SVM, we see that the performance differs greatly between classes. While the classifier performs well on the runways that are used most frequently (Kaagbaan (06/24) and Polderbaan (36L/18R)) or almost never at all (Oostbaan (04/22)), it performs poorly on runways that vary in the amount of use (Zwanenburgbaan (36C/18C), Buitenveldertbaan (09/27) and Aalsmeerbaan (36R/18L)). In other words, constant use of the runways improves their predictability. A similar pattern is found for the Logistic Regression classifier.

Runway	Precision	Recall	F_1 -score
04	0.64	0.30	0.41
06	0.54	0.80	0.65
09	0.24	0.06	0.10
18C	0.50	0.10	0.17
18L	0.42	0.10	0.17
18R	0.60	0.85	0.71
22	0.78	0.20	0.32
24	0.59	0.82	0.68
27	0.40	0.34	0.37
36C	0.00	0.00	0.00
36L	0.55	0.85	0.67
36R	0.32	0.09	0.14

Table 4.5: Precision, Recall and F_1 -scores for Logistic Regression

Hence, performance differs greatly between classes. The reason for this is that the dataset is unbalanced, meaning that not all classes contain the same number of observations. This becomes very clear from the confusion matrices. It is not always possible to recognise imbalance from the main results. In unbalanced datasets, the model might be biased towards the majority class and could be reporting various values of performance measure for different classes.

5 Discussion and Conclusion

The present study focuses on introducing a real-time decision aid to improve ATC operations. This decision-aid will assist in predicting the most suitable runway allocation for each arriving and departing flight at AAS. In order to achieve this goal, the following research questions are considered:

• Which features influence runway allocation?

To answer the first research question, the existing body of literature is explored in Section 2. Weather (primarily cloud ceiling, visibility, wind speed and wind direction), traffic demand and noise abatement all prove to be important for the ATC decision process. Whereas weather and traffic data are embedded in the current dataset, it was not possible to obtain data on noise abatement or noise measurements for AAS. However, it was possible to add some novel features to the model that have not been used before in the prediction of runway allocation. After consulting with professionals in the field, it turned out that maintenance had great influence on runway allocation, as sometimes runways are not available to ATC due to maintenance and replacements. Therefore, this feature was added as a predictor to the model. Other novel features that were included in the analyses were type of airplane and airplane weight, as the runways at AAS have different widths and lengths and not all of them are able to process the entire spectrum of airplane types.

• To what extent do the existing features influence runway allocation and which features are most influential?

The EDA in Section 3.1.1 makes it clear that airplane weights impact runway allocation: almost all light airplanes are allocated to the Oostbaan (04/22) for both arrival and departure. In Section 3.3.2 an ablation study is described to find the most predictive features for the target Runway. In general, wind direction (Wind.discrete), maintenance at the Buitenveldertbaan (Main.B), and daily temperature (Temp.daily) are the most important features according to the two classifiers used in this study, i.e., linear kernel SVM and Logistic Regression. While Wind.discrete was already identified as an important predictor by the existing literature, the results for Main.B and Temp.daily are novel. For the Logistic Regression classifier, the feature for maintenance at the Oostbaan (Main.O) is also an important predictor. However, omitting single features did not generate substantial differences in CA. Therefore, another ablation study was executed, this time to omit feature subsets. The largest subset, Weather, was most predictive for both classifiers. It enhances the accuracy of the predictions by approximately 4%. Its runner-up, Maintenance, turns out to be an important predictor for runway allocation at AAS as well, as it increases the CA of the model by approximately 2%.

• Based on the existing features, can predictions be made about runway allocation?

The third research question is answered in Section 4, where predictions are made on the test set. The linear kernel SVM and the Logistic Regression classifier are used to predict runway allocation for arriving and departing flights at AAS. Although by and large the classifiers perform rather similar, the Logistic Regression classifier consistently performs slightly better. The final experiment on the test set yields a CA score of 0.55 for the linear kernel SVM and 0.56 for the Logistic Regression classifier. Compared to previous literature, the classifiers perform rather poorly. However, this is a trivial point as the predictive performance of a model depends heavily on the available data: number of labels, balance between classes and available features and instances.

• Do these predictions yield better results than the majority baseline classifier?

The answer to the fourth research question can be found in Section 4.2. A majority classifier, which predicts the most frequently used runway direction for each flight movement, was used as a baseline. In the current study, the baseline majority classifier predicts runway 24 for all flight movements and has a CA of 0.21. Thus, the CA performance of both the linear kernel SVM (0.55) and Logistic Regression (0.56) models are well beyond the majority baseline classifier, showing that they indeed perform better than majority predictions.

5.1 Limitations

There are several limitations to the current study. For starters, the time spans in the dataset may be insufficiently detailed. In the current dataset, weather, traffic and maintenance averages were measured on a daily basis. For a complex environment as ATC, instant up-to-date data is of utmost importance. Because weather, traffic and maintenance can change frequently during the course of a day, it is best to collect data with shorter time spans, such as 10 minutes. Especially at AAS, with its unstable weather conditions, up-to-date weather information is crucial (Hesselink and Nibourg, 2011). Avery and Balakrishnan (2015) used 15-minute time intervals in their measurements and that may be one of the reasons their model achieved higher performance.

Second, the current dataset does not contain all the features that were deemed to be predictive for runway allocation in the literature. Most importantly, it was not possible to obtain data about noise abatement or data from sound level meter to map the relationships between noise pollution of individual flights and runway allocation.

Another flaw of the current dataset concerns the moderate reliability of the maintenance data. While maintenance proved to be an important predictor, it is expected to perform even better when its reliability increases. In the current dataset, maintenance planning was used to estimate the days on which there was maintenance work on a particular runway. However, practice shows that maintenance is sometimes delayed due to weather circumstances. Also, it would be desirable to distinguish between small and large maintenance projects. Small projects were not included in the planning but still can put a runway out of use for a couple of hours — something ATC has to take into account. Thus, more frequent and detailed data would be beneficial to the maintenance features as well.

A final limitation concerns the methods that were used. Classification algorithms like the linear kernel SVM and Logistic Regression try to learn the relationships between historical situations and best allocation decisions. However, they do not account for the variability in the quality of the decisions. Different decision-makers, who have varying experience, preferences and rationales will decide differently when put in the same situation. This unobserved heterogeneity introduces bias in the model estimates as it is assumed that past decisions of ATC are best practice.

5.2 Contribution to the Existing Framework

The current study contributes to the existing body of literature in several ways. The first is the use of the on-line SGD approach to discriminative learning of linear classifiers. While many previous approaches to the runway allocation problems are computationally very expensive, the SGD allows us to process large amounts of data quickly and in real-time. This enhances its practical applicability, as air traffic controllers often have to make their runway allocation decision in a matter of minutes.

The second way is that attention is directed to the use of the linear kernel SVM and Logistic Regression classification algorithms to study the runway

allocation problem. Like discrete-choice models and neural networks, these classifiers are useful in learning the mappings between an object and a class label. Although based on simple ideas, SVM and Logistic Regression often lead to high performances as the models are still complex enough to handle real-world applications (Hearst, 1998). The current method recognises that there is no optimal solution for the decision in the ATC environment, due to sometimes conflicting interests (e.g., the solution that produces the least amount of noise annoyance might not always be the safest). Whereas in optimisation techniques assumptions about preferences have to be made, classification algorithms like linear kernel SVM and Logistic Regression map according to historical decisions.

Another way in which this study contributes is the use of a large amount of data. Although not very detailed, the dataset used for this research consists of all 3.5 million flights at AAS from over the course of eight years (January 2008 to March 2016). By comparison, Ramanujam and Balakrishnan (2015) use two years' worth of data and Hesselink and Nibourg (2011) have done their research with just a single year of historical data.

The fourth way in which this research is novel is that it introduces features that have not been previously used before to model runway allocation, namely type of airplane, airplane weights, and maintenance planning. Our experiments indeed show that maintenance planning and airplane weights contribute to the predictive ability of the classifiers used, as maintenance and airplane weights increase the CA of the models by approximately 2% and 1 to 2%, respectively.

The final way is its focus on individual runway allocation, instead of runway configurations (combinations of runways). This can be very useful in situations where a configuration consists of more than one runway.

5.3 Future Research

Future research addressing runway allocation at airports should first and foremost focus on obtaining high frequency and detailed data. Best practice would be to include measurements covering 10-minute intervals to model the most recent situation at AAS. The availability of these kind of measures will increase the models ability to reveal uncovered patterns in the data and improve their performances. Also, including predictors related to noise abatement — missing from the current dataset — and reliable maintenance data would be a valuable addition.

Another suggestion for future research is to compare the performance of the model when trained on forecasts, for example weather forecasts or scheduled demand, instead of observed data. This has previously been done in the paper by Avery and Balakrishnan (2015) and although it decreased the performance of the model, it did generate a better idea of how the model would perform in real situations, when only prospect data is available. This insight will be very useful for ATC, as it gives a better idea of the usefulness of these models in real-life application.

5.4 Implications for the Field

Unfortunately, our results cannot be directly translated to the actual situation at AAS because of some of the key assumptions and simplifications in the design of the model. As stated earlier in Section 1.1.2, flawed tools can actually hurt ATC capacity, rather than enhance it. Therefore, reliable, high frequency data should be a top priority when trying to model the ATC decision process.

However, as the model performed quite well for some of the classes (the most consistently used runways), it yields a promising perspective for future research. If the quantity and quality of data increases, runway allocation can most definitely be predicted by the model. Therefore, it is likely that the efficiency of ATC at AAS can be improved by predicting runway allocation, something that has become very important in recent years due to the increasing capacity constraints at airports around the globe.

Bibliography

- Avery, J. and Balakrishnan, H. (2015). Predicting airport runway configuration. 13th USA/Europe Air Traffic Management Research and Development Seminar.
- Bennell, J. A., Mesgarpour, M., and Potts, C. N. (2011). Airport runway scheduling. 4OR Quarterly Journal of Operations Research, 9(2):115–138.
- Bertsimas, D., Frankovich, M., and Odoni, A. (2011). Optimal selection of airport runway configurations. *Operations Research*, 59(6):1407–1419.
- Busquets, J. G., Alonso, E., and Evans, A. D. (2015). Application of data mining in air traffic forecasting. 15th AIAA Aviation Technology, Integration, and Operations Conference.
- Cole, R. E., Green, S., Jardin, M., Schwartz, B. E., and Benjamin, S. G. (2000). Wind prediction accuracy for air traffic management decision support tools. 3rd USA/Europe Air Traffic Management Research and Development Seminar.
- Gilbo, E. P. (1993). Airport capacity: Representation, estimation, optimization. IEEE Transactions on Control Systems Technology, 1(3):144–154.
- Hearst, M. A. (1998). Support vector machines. *IEEE Intelligent Systems*, 13(4):18–28.
- Heblij, S. J. and Wijnen, R. A. A. (2008). Development of a runway allocation optimisation model for airport strategic planning. *Transportation Planning and Technology*, 31(2):201–214.
- Hesselink, H. and Nibourg, J. (2011). Probabilistic 2-day forecast of runway use. *National Aerospace Laboratory NLR*.
- Janic, M. (2007). A heuristic algorithm for the allocation of airport runway system capacity. Transportation Planning and Technology, 30(5):501–510.
- Kuiper, B. R., Visser, H. G., and Heblij, S. (2011). Efficient use of the noise budget at Schiphol airport through minimax optimization of runway allocations. 11th AIAA Aviation Technology, Integration, and Operations Conference.
- Kumar, K., Singh, R., Khan, Z., and Indian, A. (2008). Air traffic runway allocation problem using ARTMAP (ART1). Ubiquitous Computing and Communication Journal, 3(3):130–136.

- van Leeuwen, P., Hesselink, H., and Rohling, J. (2002). Scheduling aircraft using constraint satisfaction. *Electronic Notes in Theoretical Computer Science*, 76:252–268.
- Mackay, W. E., Fayard, A.-L., Frobert, L., and Médini, L. (1998). Reinventing the familiar: Exploring an augmented reality design space for air traffic control. *Conference on Human Factors in Computing Systems -Proceedings*, pages 558–565.
- Metzger, U. and Parasuraman, R. (2001). The role of the air traffic controller in future air traffic management: An empirical study of active control versus passive monitoring. *Human Factors*, 43(4):519–528.
- Metzger, U. and Parasuraman, R. (2005). Automation in future air traffic management: Effects of decision aid reliability on controller performance and mental workload. *Human Factors*, 47(1):35–49.
- Nogami, J., Nakasuka, S., and Tanabe, T. (1996). Real-time decision support for air traffic management, utilizing machine learning. *Control Engineer*ing Practice, 4(8):1129–1141.
- Ramanujam, V. and Balakrishnan, H. (2011). Estimation of maximumlikelihood discrete-choice models of the runway configuration selection process. *Proceedings of the 2011 American Control Conference*, pages 2160–2167.
- Ramanujam, V. and Balakrishnan, H. (2015). Data-driven modeling of the airport configuration selection process. *IEEE Transactions on Human-Machine Systems*, 45(4):1–10.
- Schiphol Group (2016). Verwachte groei Schiphol: 60 miljoen reizigers in jubileumjaar. http://www. schiphol.nl/SchipholGroup1/NieuwsPers/Persbericht/ VerwachteGroeiSchiphol60MiljoenReizigersInJubileumjaar.htm.
- Stojković, G., Soumis, F., Desrosiers, J., and Solomon, M. M. (2002). An optimization model for a real-time flight scheduling problem. *Transportation Research Part A: Policy and Practice*, 36(9):779–788.
- Vapnik, V. N. (1995). The nature of statistical learning theory.
- Zhang, R. and Kincaid, R. (2014). Robust optimization model for runway configurations management. *International Journal of Operations Research* and Information Systems, 5(3):1–25.

A Dataset description

Below you will find a detailed description of the features in the dataset used to perform this research.

Date	The date that a particular flight occurred, ranges from 2008-01-01 to 2016-03-01 (yyyy-mm-dd)
Weekday	The day of the week, string feature with seven possible values
Weekend	Whether the day of the week is either a Saturday or Sunday or another day of the week, binary feature (1=Yes, $0=No$)
Season	The season, string feature with values summer, fall, winter and spring
Icao.type	Type of airplane defined by the ICAO, e.g. A320
Act.description	Full name of the type of airplane, e.g. AIRBUS A320-2
MTOW	Maximum take-off weight of the plane (in tons of kilos), discrete variable with a range from 1 to 640
Icao.wtc	Airplane weight class defined by the ICAO, discrete fea- ture with possible values light, medium, heavy and su- per heavy
Arr.dep	Whether the flight was an arrival or a departure flight, binary feature (A=arrival, D=departure)
Wind.direc	Vector mean wind direction in degrees (360=north, 90=east, 180=south, 270=west, 0=calm/variable), discrete variable
Wind.speed.vec	Vector mean windspeed (in 0.1 m/s), continuous feature with values between 1 and 153

Wind.speed.daily	Daily mean windspeed (in 0.1 m/s), continuous feature with values between 9 and 154
Wind.speed.max	Maximum hourly mean windspeed (in 0.1 m/s), continuous feature with values between 20 and 220 $$
Wind.speed.min	Minimum hourly mean windspeed (in 0.1 m/s), continuous feature with values between 0 and 140 $$
Wind.gust.max	Maximum wind gust (in 0.1 m/s), continuous feature with values between 40 and 320 $$
Temp.daily	Daily mean temperature in (0.1 degrees Celsius), continuous feature with values between -115 and 264 $$
Temp.min	Minimum temperature (in 0.1 degrees Celsius), continuous feature with values between -188 and 201
Temp.max	Maximum temperature (in 0.1 degrees Celsius), continuous feature with values between -56 and 337
Temp.10.min	Minimum temperature at 10 cm above surface (in 0.1 degrees Celsius), continuous feature with values between -215 and 191
Sun.dur	Sunshine duration (in 0.1 hour) calculated from global radiation (-1 for <0.05 hour), continuous feature with values between 0 and 154
Sun.dur.prct	Percentage of maximum potential sunshine duration, continuous feature with values between 0 and 94
Radiation	Global radiation (in $\rm J/cm^2),$ continuous feature with values between 20 and 3070
Precip.dur	Precipitation duration (in 0.1 hour), continuous feature with values between 0 and 240 $$
Precip.daily	Daily precipitation amount (in 0.1 mm) (-1 for < 0.05 mm), continuous feature with values between -1 and 605
Precip.max	Maximum hourly precipitation amount (in 0.1 mm) (-1 for <0.05 mm), continuous feature with values between -1 and 275
Sea.press.daily	Daily mean sea level pressure (in 0.1 hPa) calculated from 24 hourly values, continuous feature with values between 9729 and 10449

Sea.press.max	Maximum hourly sea level pressure (in 0.1 hPa), con- tinuous feature with values between 9768 and 10469
Sea.press.min	Minimum hourly sea level pressure (in 0.1 hPa), continuous feature with values between 9615 and 10434 $$
Visibility.min	Minimum visibility; 0: < 100 m, 1:100-200 m, 2:200- 300 m,, 49:4900-5000 m, 50:5-6 km, 56:6-7 km, 57:7-8 km,, 79:29-30 km, 80:30-35 km, 81:35-40 km,, 89: >70 km)
Visibility.max	Maximum visibility; 0: <100 m, 1:100-200 m, 2:200-300 m,, 49:4900-5000 m, 50:5-6 km, 56:6-7 km, 57:7-8 km,, 79:29-30 km, 80:30-35 km, 81:35-40 km,, 89: >70 km)
Cloud.daily	Mean daily cloud cover (in octants, 9=sky invisible), discrete feature
Humidity.daily	Daily mean relative atmospheric humidity (in percents)
Humidity.max	Maximum relative atmospheric humidity (in percents)
Humidity.min	Minimum relative atmospheric humidity (in percents)
Evapo	Potential evapotran spiration (Makkink) (in 0.1 mm), continuous feature with values between $0~{\rm and}~57$
Wind.discrete	Wind direction, string feature with values north-east, north-west, south-east and south-west
Beaufort	Beaufort scale for wind speed, discrete feature with values from 1 to 7 $$
Vmc.imc	Whether there are VMC (visibility \geq 5 km) or IMC (visibility $<$ 5 km) at AAS, binary feature
Main.A	Whether or not there is maintenance work at runway Aals meerbaan, binary feature (1=Yes, 0=No)
Main.B	Whether or not there is maintenance work at runway Buitenveldertbaan, binary feature (1=Yes, 0=No)
Main.K	Whether or not there is maintenance work at runway Kaagbaan, binary feature (1=Yes, 0=No)
Main.O	Whether or not there is maintenance work at runway Oostbaan, binary feature (1=Yes, 0=No)

Main.P	Whether or not there is maintenance work at runway Polderbaan, binary feature $(1=Yes, 0=No)$
Main.Z	Whether or not there is maintenance work at runway Zwanenburgbaan, binary feature $(1=Yes, 0=No)$
Last.year	Total number of flights on the same date last year, discrete feature that ranges from 2 to 1445
Yesterday	Total number of flights on the day before, discrete feature that ranges from 2 to 1461
Runway	Target feature. Runway code, string feature with twelve possible values
	Exploratory Data Analysis
Name	The full name of the runway, string feature with six possible values
Total.flights	Total number of flights per day, discrete feature with values between 2 and 1461

B Results ablation study

Omitted feature	CA	Omitted feature	CA
Wind.discrete	0.518644	Sun.dur.prct	0.529281
Main.B	0.520454	Cloud.daily	0.529405
Temp.daily	0.521245	Main.A	0.529533
Yesterday	0.523031	Main.P	0.529964
Weekend	0.524226	Precip.max	0.530043
Sun.dur	0.526015	Wind.speed.min	0.530212
Wind.speed.max	0.526230	Wind.speed.daily	0.530600
Season	0.526341	Wind.direc	0.530888
Sea.press.max	0.526931	Temp.min	0.531209
Temp.10.min	0.526933	Arr.dep	0.531598
Main.Z	0.527032	MTOW	0.532071
Humidity.daily	0.527246	Temp.max	0.532082
Wind.gust.max	0.527275	Precip.dur	0.532604
Wind.speed.vec	0.527900	Humidity.max	0.532930
Visibility.min	0.527919	Evapo	0.533895
Precip.daily	0.527920	Visibility.max	0.534054
Icao.wtc	0.527929	Last.year	0.534430
Main.K	0.528023	Weekday	0.534507
Main.O	0.528399	Sea.press.min	0.534723
Radiation	0.529048	Beaufort	0.535158
Sea.press.daily	0.529055	Vmc.imc	0.538182
Humidity.min	0.529165	None	0.538872
	1. C	1 1	1 1

The following tables show the effect on CA when omitting individual features from the dataset.

Table B.1: CA results for ablation study with linear kernel SVM

Omitted feature	CA	Omitted feature	CA
Main.O	0.536601	Cloud.daily	0.545799
Wind.discrete	0.539650	Sea.press.daily	0.545967
Main.B	0.539825	Precip.daily	0.546151
Icao.wtc	0.541084	Evapo	0.546188
Temp.daily	0.542107	Yesterday	0.546257
Beaufort	0.542227	Vmc.imc	0.546429
Main.K	0.542908	Sun.dur	0.546448
Wind.speed.daily	0.543233	Sea.press.max	0.546515
MTOW	0.543298	Main.P	0.546517
Precip.max	0.543444	Last.year	0.546543
Wind.speed.vec	0.544065	Radiation	0.546663
Temp.10.min	0.544256	Main.A	0.546744
Sea.press.min	0.544589	Wind.speed.min	0.546774
Visibility.max	0.544622	Wind.speed.max	0.546832
Wind.gust.max	0.544833	Weekday	0.546984
Sun.dur.prct	0.544868	Temp.min	0.547394
Humidity.min	0.544872	Temp.max	0.547931
Weekend	0.544916	Main.Z	0.547935
Wind.direc	0.545026	Season	0.548115
Humidity.daily	0.545166	Arr.dep	0.548383
Visibility.min	0.545285	Humidity.max	0.548632
Precip.dur	0.545629	None	0.549264

 Table B.2: CA results for ablation study with Logistic Regression

C Confusion matrices

This appendix contains the confusion matrices for the performances of the linear kernel SVM and Logistic Regression classifiers on the test set.

								Predicted						
		04	06	09	18C	18L	18R	22	24	27	36C	36L	36R	\sum
	04	693	331	12	7	16	32	298	58	24	4	812	12	2299
	06	92	37528	3	312	0	5408	204	6	2484	222	43	1318	47620
	09	8	100	880	2	969	0	11	2992	0	7	7555	0	12524
	18C	2	1037	0	1160	608	36263	384	1662	683	184	301	424	42708
	18L	1	0	1403	13	5288	0	32	45527	0	404	4026	0	56694
	18R	3	4623	0	737	0	97469	367	0	2705	0	0	965	106869
True	22	166	175	45	18	181	8000	2942	2074	109	15	400	20	14145
	24	3	89	2579	53	5401	1082	171	90056	155	147	14514	31	114281
	27	5	3477	0	445	46	20534	453	2191	7142	192	267	765	35517
	36C	7	2102	784	48	141	1629	53	761	2055	106	14080	614	22380
	36L	21	0	806	1	715	0	25	5274	0	73	50020	0	56935
	36R	44	13954	0	151	0	1407	107	0	2086	224	0	1559	19532
	\sum	1045	63416	6512	2947	13365	171824	5047	150601	17443	1578	92018	5708	531504

Table C.1: Confusion matrix for the linear kernel SVM classifier

								Predicted						
		04	06	09	18C	18L	18R	22	24	27	36C	36L	36R	\sum
	04	700	369	35	9	13	23	292	87	19	0	737	5	2289
	06	47	37914	1	687	1	4366	72	7	2870	0	43	1612	47620
	09	10	100	788	0	675	0	6	3949	0	0	6996	0	12524
	18C	1	1610	75	4249	1194	32280	29	1181	1574	0	301	214	42708
	18L	1	0	348	0	5893	0	14	46767	0	0	3671	0	56694
	18R	0	6402	0	2073	0	90637	94	0	7064	0	0	599	106869
True	22	238	130	20	385	212	5440	2819	2163	2352	0	375	11	14145
	24	8	208	706	37	5272	786	110	93314	374	3	13455	8	114281
	27	3	5343	0	912	166	13987	132	2104	12092	2	232	544	35517
	36C	26	2747	201	54	43	1173	15	1704	2005	0	13804	608	22380
	36L	32	0	1059	0	615	0	19	6930	0	2	48278	0	56935
	36R	24	14862	0	67	0	1250	21	0	1617	0	0	1691	19532
	\sum	1090	69685	3233	8473	14084	149942	3623	158206	29967	7	87892	5292	531504

Table C.2: Confusion matrix for the Logistic Regression classifier