DEGREE PROJECT IN COMPUTER SCIENCE AND ENGINEERING,
SECOND CYCLE, 30 CREDITS
STOCKHOLM, SWEDEN 2019

Evaluation of machine learning algorithms for customer demand prediction of in-flight meals

MATTEUS HAST

KTH ROYAL INSTITUTE OF TECHNOLOGY
SCHOOL OF ELECTRICAL ENGINEERING AND COMPUTER SCIENCE

Evaluation of machine learning algorithms for customer demand prediction of in-flight meals

MATTEUS HAST
mhast@kth.se

Master thesis in Computer Science
Date: July 2, 2019
Supervisor: Somayeh Aghanavesi
Examiner: Erik Fransén
Swedish title: Utvärdering av maskininlärningsalgoritmer för prediktering av konsumentefterfrågan för måltider under flygning
School of Computer Science and Communication

Abstract

This study aims to evaluate multiple Machine Learning Algorithms (MLAs) for estimating the customer demand of in-flight meals. As a result of the review of related works, four MLAs were selected, namely Linear Regression (LR), Support Vector Regression (SVR), Extreme Gradient Boosting (XGBoost) and a Multilayer Perceptron Neural Network (MLP). The study investigates which MLA is best suited for the problem at hand and which features are most influential for customer demand prediction of in-flight meals. Focus is put on finding applicable MLAs and on evaluating, comparing and tweaking the parameters of the MLAs to further optimise the selected models. The available data set comes from a single airline company and consists mainly of flights with a short to medium long flight duration time.

The results show that the four evaluated models, LR, SVR, XGBoost and MLP performs with no significant difference against one another and are comparable in their performance in regard to estimation accuracy with results close to each other's. However, the SVR model underperforms in regard to model fitting and prediction time in comparison towards the remaining three models. Furthermore, the most important feature for customer demand prediction of in-flight meals is the scheduled flight duration time.

Sammanfattning

Syftet med den här studien är att utvärdera ett flertal maskininlärningsalgoritmer för prediktering av konsumentefterfrågan för måltider under flygning. Undersökningen över tidigare arbeten utförda i liknande fält resulterade i att fyra maskininlärningsalgoritmer blev valda, nämligen linjär regression, stödvektormaskin för regression, Extreme Gradient Boosting och ett flerlagersperceptron-neuronnät. Studien utforskar vilken maskininlärningsalgoritm som är bäst anpassad för att prediktera problemet samt vilka egenskaper i datat som är mest inflytesrika när det handlar om att prediktera konsumentefterfrågan av måltider under flygning. Fokus ligger på att finna applicerbara maskininlärningsalgoritmer och på att utvärdera, jämföra samt på att justera parametrarna i syfte till att optimera modellerna. Den tillgängliga datan härstammar från ett enstaka flygbolag och består mestadels av korta och mediumlånga flyg.

Resultatet påvisar att de fyra modellerna, linjär regression, en stödvektormaskin för regression, Extreme Gradient Boosting och ett flerlagersperceptron-neuronnät presterar utan någon signifikant skillnad gentemot varandra och är jämförbara i deras prestation i avseende till predikteringprecision med liknande resultat. I avseende till modellanpassnings- och predikteringstid underpresterar dock stödvektormaskinen avsevärt i jämförelse med de resterande tre modellerna. Resultatet visar även att den viktigaste egenskapen i datat för prediktering av konsumentefterfrågan av måltider under flygning är den schemalagda flygtiden.

Acknowledgements

I would like to thank Objective Solutions Sweden AB for hosting me, providing me with necessary equipment, and for the opportunity to conduct my research with them. Special and sincere thanks to my supervisor Jesper Ljungehed, who provided me with valuable input and guidance throughout the degree project and to Adriaen Kamenický for making it possible.

Finally, I would like to thank my wife Shilan Hast, for your continuous support during all of my academic years. You have motivated and influenced me more than you can imagine and I would not be where I am today without you.

Contents

Glossary

ANN Artificial Neural Network (ANN) is a computing system inspired by biological neural networks.

DT Decision Tree (DT) is a tree like model of decisions and their outcomes.

FNN Feedforward Neural Network (FNN) is the most basic neural network.

LOGIT Logistic Regression (LOGIT) is a supervised regression algorithm.

LOWESS Locally Weighted Scatterplot Smoothing (LOWESS) is a regression-smoothing technique, commonly used for visualising trends and patterns in data.

LR Linear Regression (LR) is the one the most basic supervised regression algorithm.

MAE Mean Absolute Error (MAE) is an error metric.

MAPE Mean Absolute Percentage Error (MAPE) is an error metric.

MLA Machine Learning Algorithm (MLA) is a learning algorithm for an intended task with an input and output.

MLP Multilayer Perceptron (MLP) is a class of FNN with at least one input layer, one hidden layer and one output layer.

MSE Mean Squared Error (MSE) is an error metric.

R^2 score Coefficient of determination (R^2 score) is a metric of the goodness of a fit of a model.

RBF Radial basis function (RBF) refers to a kernel commonly used in SVMs and SVRs.

RF Random Forest (RF) is a supervised ensemble regression algorithm of multiple regression trees.

RMSE Root Mean Square Error (RMSE) is an error metric.

RT Regression Tree (RT) is a Decision Tree in which its target value can take continuous values.

SVM Support Vector Machine (SVM) is a supervised classification algorithm.

SVR Support Vector Regression (SVR) is an SVM but for supervised regression.

XGBoost Extreme Gradient Boosting (XGBoost) is a supervised ensemble regression algorithm.

Chapter 1

Introduction

A passenger taking a commercial plane is subject to a variety of in-flight services, such as food, drinks and games. A study conducted in 2009 by Myungsook An and Yonghwi Noh examines the impact of in-flight service quality on airline customer satisfaction and loyalty. In the study, An and Noh find that there are five service quality factors of importance for all passengers taking a flight: responsiveness and empathy, food quality, alcoholic beverage, non-alcoholic beverage, and reliability [1]. One of these factors concerns the interaction with the airline personnel whilst the rest concerns the in-flight food and drink services. This shows the importance for an airline company of delivering satisfying such services. One important factor amongst these five are “reliability”, a passenger should be able to purchase a drink if they are thirsty and an in-flight meal if they are hungry. A problem, however, arises when deciding the correct amount of drinks and in-flight meals to load, especially for the fresh food in-flight meal service.

Airline companies operating within European borders are bound by European Union (EU) regulations, including regulations regarding the cold food chain. It is required by EU regulations that airline companies throw away food that have broken the cold chain [2], which refers to food that has at some point been taken out of the refrigerator or freezer. Storing meals in-flight counts as breaking the cold chain and as such, all remaining in-flight meals must be thrown away when the plane arrives at its destination. The dilemma is as such, to either load large amounts of meals with the risk of food waste or to load at lesser capacities to minimise food waste but with the risk of customer dissatisfaction.

During a meeting with an airline company in February 14th, 2019 [3], it was pointed out that historically, the decision on how many meals to load has been decided iteratively. Numerous flights have flown throughout the years and by each one the estimation has improved. One problem, according to the airline company, arises when new directives arrive either to manage environmental challenges or to increase customer satisfaction [3]. The dilemma is thus still the same, how can an airline company maximise customer satisfaction all the while minimising food waste, hence satisfy both the environmental and customer variables.

Machine learning could offer a solution to the dilemma. Machine learning and data driven approaches are becoming important and common in many areas. Smart spam filter classifiers protect us from potentially harmful emails by learning from massive amounts of previous emails and user feedback; fraud detection protect banks from malicious attackers; smart vehicle systems not only increases safety for drivers, pilots and pedestrians but also decreases travel time and fuel expenses; sales forecasting aids stores and product chains in increasing sales and decreasing expenses [4, 5, 6, 7, 8, 9]. The main two success factors behind these systems are the usage of effective models that are able to capture the data dependency and scalable learning systems that teaches the model from large data sets [10, 11]. A Machine Learning Algorithm (MLA) is a learning system for creating a model which can predict a value based on its learning from previous examples. For example, predicting the amount of in-flight meals to load onto a flight from Stockholm to Oslo on a Saturday morning based on consumed amount of in-flight meals of previous such flights.

This study aims to solve the problem of finding the optimal amount of in-flight meals to load onto a commercial flight by evaluating multiple MLAs. The study will compare multiple MLAs and the purpose of this study is that by estimating the optimal amount of in-flight meals, little to no meals will be left over to be thrown away at arrival. The significance of the research should not only be measured by a single airline company, this research has a high applicability of similar use cases in both the airline industry as well as other industries. The applied method offers an evaluation of different MLAs to find the most optimal solution for this problem. This is to serve as a guideline for evaluations in related fields and as a basis for further research in predicting customer demand of in-flight services.

1.1 Research Question

Assessing and evaluating multiple MLAs on a customer demand prediction problem has proven to be beneficial in previous studies [6, 12, 13, 14, 15] and serves as an important foundation for further research. In order for an airline company to satisfy both the environmental and customer variables one needs to load as close to a perfect amount of in-flight meals as possible. By finding the optimal amount of in-flight meals, an airline company will not only minimise food waste but also increase customer satisfaction and sales revenue. The question then arises if it would be possible with the help of an MLA and historical data of flights to predict this amount of meals, specific for a certain flight.

As such, this study is an evaluation of machine learning methods to estimate the customer demand of in-flight meals and the questions this study intends to answer are:

- Which of the attributes in the available data are most relevant for customer demand prediction of in-flight meals?
- Which of the evaluated machine learning methods are best suited to solve the problem?

1.2 Scope

The research is limited to data from a single anonymised airline company with parameters concerning flight information, passenger booking class and in-flight sales information. Therefore, conclusions drawn might only be applicable to the particular data set. Subcategories of in-flight meals e.g. sandwiches, salads and so forth are not considered for this study. The models will be presented one meal category and this limitation was necessary due to a limited time frame and was decided in collaboration with the airline company [3].

Furthermore due to a limited time frame, the MLAs to be evaluated has been limited to four regression algorithms, namely Linear Regression (LR), Support Vector Regression

(SVR), Extreme Gradient Boosting (XGBoost) and Multilayer Perceptron (MLP) Feedforward Neural Network (FNN). This limitation was made to ensure depth and relevance. The four selected regression algorithms were selected because of their results in similar demand prediction and sales forecasting studies [6, 16, 17, 18, 19].

1.3 Structure of the Paper

The rest of this paper is organised as follows: in Chapter 2, the related work in the field is studied. In Chapter 3, the theoretical framework of this thesis is presented. Relevant theory is presented in this chapter. In Chapter 4, the chosen method and implementation details are described. In Chapter 5, the result of the conducted case study is presented, and in Chapter 6 conclusions drawn from the result is presented. This thesis concludes in Chapter 7 with final remarks on the findings and the contributions. Suggestions for future work as well as a critical review of the thesis is presented in this chapter.

Chapter 2

Related Work

This chapter presents and examines the previous related studies in the area and intends to motivate why certain methods and concepts were chosen, with respect to limitations and the available data.

2.1 Evaluations of Machine Learning Methods

In 1997, Gregory F. Cooper et al. performed an evaluation on detecting pneumonia in patients and compared the results of eight different methods. Their results showed that out of the eight methods, Neural Networks, Decision Trees (DTs) and Logistic Regression (LOGIT) performed best. All eight MLAs were however within an 1% error rate and the top three were within a 0.5% error rate. Their conclusion follows that their MLAs would be able to predict even better if more data were to be used in the training of the model [12].

Similarly, in 2015 an evaluation of four MLAs was conducted by Henrik Almér on the prediction of fuel consumption. Almér comes to the conclusion that SVR, Random Forest (RF) and an MLP FNN provides the most accurate predictions with no significant difference amongst themselves. These findings agree with Cooper et al. as Neural networks and tree algorithms are shown to perform amongst the best. Furthermore, in the study by Almér, LR is used as baseline/benchmark for the more advanced MLAs [6].

As opposed to Cooper et al. and Almér, a study by V. Rodriguez-Galiano et al. from 2015 shows that RF outperforms both an Artificial Neural Network (ANN) and a Regression Tree (RT). In their study for mapping mineral prospectivity, four MLAs are evaluated, namely an ANN, Support Vector Machine (SVM), RF and RT. Their results show that for mapping mineral prospectivity, a RF model outperforms ANN models, SVM models and RT models. The study concludes in that both RTs and RF model parameters can be interpreted to gain valuable insight. Whereas this was not seen at neither ANNs nor SVMs [13].

An evaluation of MLAs for classification conducted by J. Ling and J. Templeton predicted regions of high Reynolds averaged Navier Stokes uncertainty. They compare the result of three different MLAs, namely SVMs, Adaboost DTs and RFs. Similar to Rodriguez-Galiano et al., they conclude that for their classification problem, RF had the best combination of good performance and easy implementation [15].

A study by Raymond Salvador et al. on evaluating MLAs for classifying structural features for optimal MRI-base diagnostic prediction in psychosis compared a wide range of commonly used MLAs for classification. As opposed to the studies by Cooper et al., Rodriguez-Galiano et al. and Ling and Templeton, their results show that all MLAs evaluated performed with no significant difference amongst themselves. Their study concludes in that the selection of feature type is of more importance than the selection of MLA [14].

Out of these five studies a wide range of different MLAs have been evaluated and compared against one another. The top performers out of these are namely LOGIT, SVMs, SVRs, RTs, DTs, RFs and ANNs. Out of these seven MLAs, four were selected for this study based on their result in previous research and their applicability in predicting customer demand of in-flight meals. As such, an SVR, a tree ensemble algorithm (RTs, RF) and an ANN were selected. Furthermore, based on the findings in the study by Almér on evaluating four regression MLAs [6], LR will be used as a baseline for comparison toward the other three more advanced MLAs.

2.2 Estimations on Customer Demand

In order to find applicable MLAs for estimating customer demand of in-flight meals, previous research on applying different MLAs on predicting customer demand and on sales forecasting are considered and examined. This combined with the studies on MLA evaluations serves as a basis for the selection of MLAs and performance metrics.

A study by Ahmet Selman Bozkir and Ebru Akcapinar Sezer on customer demand from 2011 aims to predict the food demand in food courts by DT approaches, namely Classification and Regression Tree (CART), Chi Squared Automatic Interaction Detection (CHAID) and Microsoft Decision Trees (MSDT). In the study, historic data from a period of two years of a university food court is examined and the results demonstrate that DT methodology is suitable for food consumption prediction. Furthermore, they achieve prediction accuracies up to 0.83 in Coefficient of determination (R^2 score) [20].

Similar to the study by Bozkir and Sezer, a study from 2005 by Ana Lúcia Silva and Margarida GMS Cardoso aims to predict sales forecasting instead of customer demand by tree algorithm approaches. Their study proves the possibility of predicting supermarket sales using RTs influenced by both internal and external factors. The focus of the study is on examining the importance of external environmental factors on the predictions. The results show that external factors and factors such as the stores' lack of visibility and heavy road traffic contribute especially to the decrease of sales in smaller stores [19].

Another study on sales forecasting from 2015 examines if retails sales and demand can be predicted with the help of an MLA. As opposed to the studies by Bozkir and Sezer and Silva and Cardoso, the study by Ankur Kumar Jain, Manghat Nitish Menon, and Saurabh Chandra compares the use of an XGBoost model against LR and RF. Their results show that XGBoost outperforms both LR and RF, in regards to prediction accuracy [9].

A study from 2015 by A.A. Levis and L.G. Papageorgiou evaluates SVRs for demand forecasting. Their model is based upon historical data and it is shown that their algorithm features an adaptive and flexible regression function able to identify the underlying customer demand patterns from the available training points so as to capture customer behaviour and derive an accurate forecast. Their final SVR has a prediction accuracy of 93% on all of their presented cases [21].

Further sales forecasting based on SVR was made in a study by Chi-Jie Lu in 2013. The study combines variable selection methods and SVR for constructing a hybrid sales forecasting model for computer products. The results show that the proposed hybrid sales forecasting scheme provides a better forecasting result than four other competing models in terms of forecasting error. All five models are within a Mean Absolute Percentage Error (MAPE) of 22 and the hybrid model results in a MAPE of 17.27 [22].

A study from early 2018 on sales forecasting using ANNs by Rosa Cantón Croda, Damian Gibaja, and Omar Caballero, applies an MLP for predicting the sales of a warehouse enterprise. Their findings show that learning rate variations do not significantly increase the computing time, and that the validation fails with an error less than five percent [17].

A study from 2009 by Zong Woo Geem and William E. Roper demonstrated the possibility of estimating the energy demand of South Korea using MLP FNN. In agreement with the study by Cantón Croda, Gibaja, and Caballero, their findings show that an MLP is a suitable model for predicting customer demand. The proposed model estimates the energy demand better than a linear regression model and an exponential model, in terms of Root Mean Square Error (RMSE). The proposed model could detect underlying trends, energy demand peaks and lows, which neither the linear nor exponential model could [23].

These studies further prove the applicability of the four selected MLAs (a LR, a SVR, a tree ensemble algorithm and an ANN) for customer demand prediction of in-flight meals. What remains is the selection of a tree ensemble algorithm and an ANN. In the study by Almér [6], by Cantón Croda, Gibaja, and Caballero [17] and by Geem and Roper [23], an MLP ANN was evaluated for a supervised regression learning task. Their findings point to promising results in similar studies which is why an MLP is selected for this study. Furthermore, both RTs and RF shows promising result in related studies [6, 9, 19] with the exception that in the study by Jain, Menon, and Chandra on sales forecasting for retail chains an XGBoost was evaluated compared to RF. The results in their study is what made the final choice of XGBoost as the ensemble tree algorithm for this study. Furthermore, the findings in the study on sales forecasting in 2015 by Jain, Menon, and Chandra and on energy demand in South Korea by Geem and Roper further prove the applicability of selecting LR as a baseline for comparison.

Chapter 3

Theoretical Framework

This chapter presents the relevant theory and background necessary to comprehend the presented methods and evaluations. It begins with a section on the origin of the problem followed by a section on machine learning and the applied MLAs. The chapter concludes in sections on possible model performance improvement techniques and performance metrics for assessing and analysing the selected MLAs

3.1 Customer Demand in the Aviation Industry

According to An and Noh, the airline customer satisfaction and loyalty can be measured in the in-flight service quality. In a majority of all flights, there is at least two booking classes, business and economy. In the business class, according to An and Noh, there are six service quality factors of importance: alcoholic and non-alcoholic beverage, responsiveness and empathy, reliability, assurance, presentation style of food, and food quality; while for the economy class there are five: responsiveness and empathy, food quality, alcoholic beverage, non-alcoholic beverage, and reliability. Both classes include both food quality and more importantly reliability [1]. The competitive landscape of the European aviation industry has developed significantly throughout the years due to an escalation in competition and a growth of low-cost carriers. This has led to many carriers lowering flight ticket prices by minimising costs through various means. One such method of lowering cost and increasing revenue is by instead of offering a complementary meal for each passenger, airline companies are selling it [24]. It then not only becomes a question of customer satisfaction but of customer demand, to maximise customer satisfaction by meeting customer demand.

3.2 Machine Learning

Machine learning is a field in statistical analysis and Computer Science that aims to build computer programs that learn and improve their performance through experience [10]. It is a field that has grown rapidly in the last two decades, what started as laboratory curiosity has spread into something with a widespread commercial usage. Within the field of Artificial Intelligence (AI), machine learning approaches have become the method of choice. Many researchers and developers recognise the simplicity and applicability of training a system by showing it examples of desired input and output instead of manually having to program it [25].

Artificial intelligence is however not the only field where machine learning has proven to be useful, it has proven to be of great practical value in a variety of application domains. Data mining problems with large databases with implicit regularities that an automatic computer system can discover; in domains where humans might not have the knowledge needed to create effective algorithms; in domains where the program itself must dynamically adapt to changing conditions. A well-defined learning problem requires a well specified task, performance metric and source of training experience and when designing a machine learning approach, a number of design choices are involved. The type of training experience, the target function to be learned, a representation of this target function and an algorithm for learning the target function from training experience are all design choices which needs to be specified. A machine learning approach to this well-defined task is also referred to as a machine learning method [10].

There are multiple machine learning methods where the most widely used are supervised learning. Supervised learning systems generally form their predictions via a learned mapping f(x), which produces an output y for each input x (or a probability distribution over y given x). Many different forms of mapping f exist, including DTs, Decision forests, LOGIT, SVMs, ANNs, Kernel machines, and Bayesian classifiers [26], these are examples of what will be referred to as MLAs. Further types of machine learning methods are unsupervised, semi-supervised and reinforcement learning. In the case of unsupervised learning the outcome/output for the given input is unknown and in the case of semi-supervised it is a combination of both labelled and unlabelled data. In reinforcement learning, the method gets exposed to an environment where it learns by trial and error. All these learning types and the MLAs may be applied to either classification, predicting discrete values or regression which instead predicts continuous values [10]. This research will only focus on supervised regression MLAs, specifically LR, SVR, XGBoost and MLP as the data is labelled and the output a continuous value.

3.3 Machine Learning Algorithms

3.3.1 Linear Regression

LR is based on regression analysis which is a statistical technique for modelling and investigating the relationship between variables. Because of its simplicity and applicability, LR is widely used. The most basic variant is simple linear regression where there is one **response**, y and one **regressor**, x (Equation 3.1).

$$y = B_0 + B_1 x + \varepsilon \tag{3.1}$$

Another variant is multiple linear regression, where a response is dependent on multiple regressors (Equation 3.2) [27].

$$y = B_0 + B_1 x_1 + B_2 x_2 + ... + B_k x_k + \varepsilon = B_0 + B^T x + \varepsilon \tag{3.2}$$

LR can be used for estimating regressors, proving a relationship and estimating the response, which is how it is applied in this study. The ε is the error or the "noise variable" and since it is unobserved, a model is needed to be able to predict the output value y given x. Given this, the following prediction model is obtained [28]:

$$\hat{y} = \hat{B}_0 + \hat{B}_1 x_1 + ... + \hat{B}_k x_k + \varepsilon = \hat{B}^T x + \varepsilon = \sum_{n=1}^{N} x_n y_n \tag{3.3}$$

The Least-Square loss function is used to calculate $\hat{B}$, proving that $\hat{B}$ is equal to B_* and by minimising, with respect to B the cost function is obtained [28]:

$$J(B) = \sum_{n=1}^{N} (y_n - B^T x_n)^2 \tag{3.4}$$

In this study the Least-Square loss function is the only loss function that is considered.

3.3.2 Support Vector Regression

SVR is built upon SVMs, which are a popular form of supervised learning for classification. SVR uses the same principles as a SVM but applies to regression instead of classification. SVMs originates from statistical theory and was first constructed in Russia in the sixties [29]. The idea behind a SVM is to make a transformation of the input data followed by a linear separation where the decision boundary is placed to give maximal space to data points. In order to find this decision boundary, support vectors are introduced. The support vectors are data points which are lying on or inside of the margin of the decision boundary, as is depicted in Figure 3.1, where the support vectors are the circled dots.

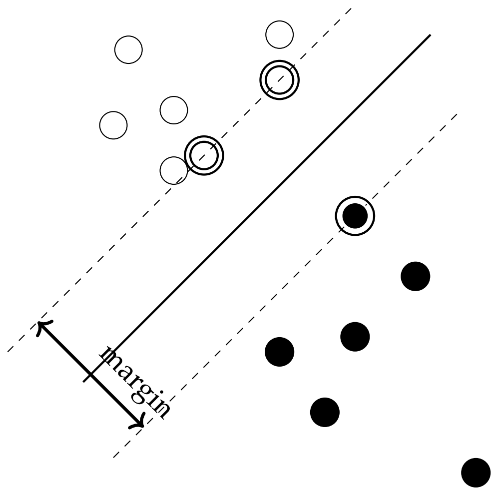


Figure 3.1: Example of an SVM. Depicting a decision boundary between two classes with support vectors.

Using something called the kernel trick, SVMs can transform its input data to higher dimensions in order to make it a better fit for classification [29]. The kernel is a function in which its input data is transformed to higher dimension and as such is capable of comprehending more advanced relationships between features and variables. Examples of the kernel function $\kappa(x, x')$ are the linear kernel $\langle x, x' \rangle$, the polynomial kernel $(\gamma\langle x, x' \rangle + c)^d$, the Radial basis function (RBF) kernel $\exp(-\gamma\|x - x'\|^2)$ and the sigmoid kernel $\tanh(\gamma\langle x, x' \rangle + c)$. The parameter γ defines how far the influence of a single training example reaches, with low values meaning "far" and high values meaning "close". The parameter d is a specified degree and c is a coefficient trading off the influence of higher-order versus lower-order terms [30].

Compared to SVMs, SVRs finds a function approximation that minimises the error and like SVMs, SVRs optimises the generalisation properties of the model [31]. Calculating the correct estimation through SVRs is best seen as an optimisation task where Lagrange multipliers are used and forming the corresponding Lagrangian [32] (Equation 3.5).

$$\hat{\theta} = \sum_{n=1}^{N}(\tilde{\lambda_n} - \lambda_n)x_n \tag{3.5}$$

Where x is the input vector, θ is the weight and $\tilde{\lambda_n}$, λ_n, $n = 1, 2, ..., N$ are the Lagrange multipliers associated with each one of the constraints. Furthermore, the bias can be obtained through the equations:

$$y_n - \theta^T x_n - \theta_0 = \varepsilon \tag{3.6}$$

$$\theta^T x_n + \theta_0 - y_n = \varepsilon \tag{3.7}$$

By obtaining θ and θ_0, prediction can now be performed. By beginning with the feature mapping κ the following is obtained [32]:

$$\hat{y}(x) = \sum_{n=1}^{N_s}(\tilde{\lambda_n} - \lambda_n)\kappa(x, x_n) + \hat{\theta}_0 \tag{3.8}$$

Which is the learned mapping $f(x)$ where y for a given x is predicted.

3.3.3 Extreme Gradient Boosting

XGBoost is a scalable end-to-end machine learning system for tree boosting which has gained praise in the recent years due to its success in many data mining challenges [11, 33]. A tree model for boosting, either a RT or a DT, is a learning model created by iteratively partitioning the available data set and fitting a simple prediction model at each step. This has the advantage of easily being visualised by a graph with a tree structure. The main difference between DTs and RTs is that in RTs a continuous value is associated with each leaf [34]. Tree boosting is an ensemble technique which refers to a model constructed of multiple weaker models, generally RTs or DTs. Figure 3.2 depicts a simple ensemble tree model, where multiple DTs are combined, and the final prediction is the sum of the prediction from each tree. The output of the example in Figure 3.2 is whether or not the customer was satisfied with a specified flight.

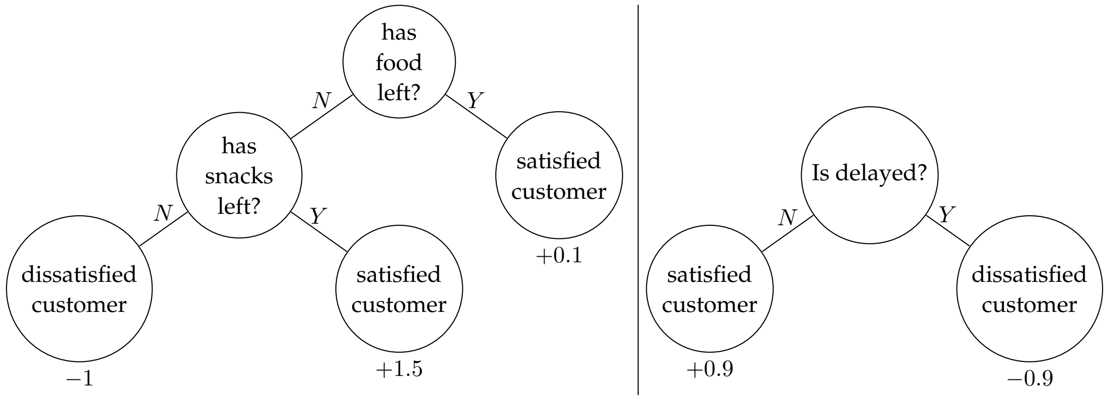
Gradient Boosting

Gradient boosting is an ensemble machine learning technique for regression and classification problems which combines weak learners, typically DTs or RTs, into a single strong learner in an additive manner [35]. For a data set with m features and n examples the following is true:

$$D = \{(x_i, y_i)\}(|D| = n, x_i \in \mathbb{R}^m, y_i \in \mathbb{R}) \tag{3.9}$$

A tree ensemble model uses K additive functions to predict the output:

$$\hat{y} = \phi(x_i) = \sum_{k=1}^{K} f_k(x_i), f_k \in F$$
$$\text{where } F = \{f(x) = w_{q(x)}\}(q : \mathbb{R}^m \to T, w \in \mathbb{R}^T) \tag{3.10}$$



f(satisfied customer) $= 1.5 + 0.1 + 0.9 = 2.5$, f(dissatisfied customer) $= -1 + -0.9 = -1.9$

Figure 3.2: Example of an ensemble tree model. The final prediction for a given example is the sum of the predictions from each tree.

Where F is the space of regression trees, also known as CART. In F, q represents the structure of each tree that maps an example to the corresponding leaf index and T is the number of leaves in the tree. Each f_k corresponds to an independent tree structure q and leaf weights w. w_i represents the value on the i-th leaf.

The ensemble model cannot be optimized using traditional methods since it includes functions as parameters and is therefore trained in an additive manner. To learn the set of functions used in the gradient tree boosting model the following objective needs to be minimised [11].

$$
\begin{gathered}
L^{(t)} = \sum_{i=1}^{n} l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(f_t) \\
\text{where } \Omega(f) = \gamma T \frac{1}{2} \lambda \|w\|^2
\end{gathered}
\tag{3.11}
$$

XGBoost is an improvement on the existing Gradient Boosting technique [9]. As in the case of gradient boosting, XGBoost is built on a tree ensemble model which is a set of classification and regression trees (CART). This is constructed in three steps.

1. Additive training. In this step the functions containing information about the structure of the tree and leaf score are defined. This is optimised using an additive strategy, by fixing what has been learned and adding a new tree at a time.

2. Model Complexity. In order to penalize certain cases, the complexity of the tree serves as a regularisation parameter.

3. Structure Score. This score contains the information on the best split conditions while taking the model complexity into account [9, 11].

3.3.4 Artificial Neural Network

ANNs have proved to be a successful learning model throughout the years, examples are in recognising speech and handwriting, autonomous cars and more. ANNs are a robust method for approximating real-valued, discrete-valued and vector-valued target functions.

ANNs have historically been inspired partly by observation of the human biological learning systems, which are built up by complex webs of interconnected neurons [10]. A neural network is a component made up of a numerous amount of neurons with synaptic links in between them. In 1943, Warren McColloch and Warren Pritts proved that given any sufficient number of neurons and adjusting appropriately the synaptic links, any computational problem can be solved [36].

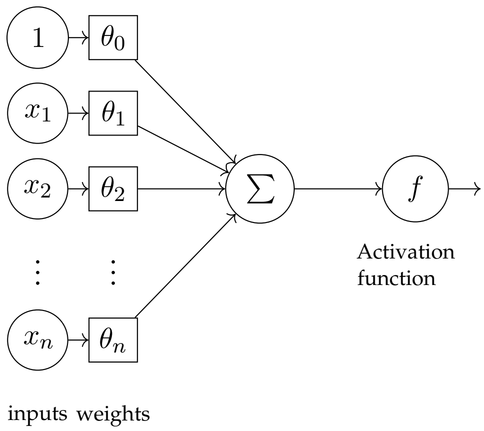


Figure 3.3: A basic neuron/perceptron architecture where the input features x are applied to the input nodes and weighted with their respective weight θ and calculated, with regards to the bias θ_0.

One common type of ANN system is based upon a unit called a perceptron (Figure 3.3). A perceptron is a unit that takes real-valued inputs, calculates a linear combination and outputs 1 or -1 depending on the result being greater than some threshold or not. One limitation of a single perceptron is that they can only express linear decision surfaces and in order to express nonlinear decision surfaces, MLP networks learned by the backpropagation algorithm are introduced [10].

Multilayer Perceptron Networks and the Backpropagation Algortihm

In this study, focus is on MLP FNN, where feedforward refers to the restriction that information only flows one way in the network, forward. A MLP network consists of at least three layers, one input layers, one or more hidden layer and one output layer, as can be seen in Figure 3.4 [10].

The backpropagation algorithm learns the weights for a MLP, given a network with a fixed set of units and interconnections. The algorithm attempts to minimise the squared error between the network output values and the target values by employing gradient descent [10]. In this study, only one hidden layer is considered for the MLP as it has been historically sufficient with one hidden layer to approximate continuous functions in most function approximation problems [37].

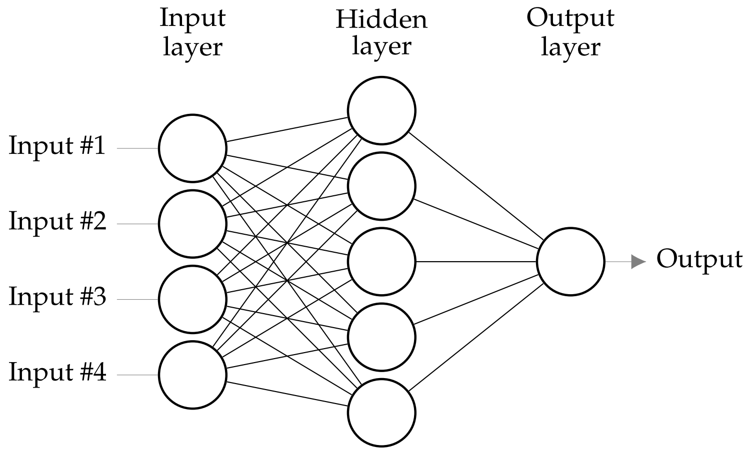


Figure 3.4: MLP with one hidden layer.

3.4 Model Performance Improvement Techniques

3.4.1 Bias and Variance

The training of a model needs to be carefully executed in order to not to consider a too rich of hypothesis space, allowing the model to overfit the training examples. This can lead to a seemingly high accuracy but with an actual low accuracy on future examples, this means that the model is biased. To ensure an unbiased model, a subset of the data should be used as future examples and not used when training the model. This refers to splitting the original data set into a train subset which is used for creating the model and a test subset for verifying the model. By selecting a too small training data set for the model, there is a risk of having a too specific model with a high accuracy on the training set but a low accuracy on the test set [10].

3.4.2 Cross Validation

Cross validation is a resampling technique used to evaluate machine learning models on a limited data set, primarily to estimate the performance of a model on unseen data and to assess the generalization ability of a particular method. Cross validation is often commonly referred to as k-fold cross validation where k acts as the technique's sole parameter. The procedure of the technique involves randomly shuffling the data set, splitting the shuffled data set into k groups and for each unique group take that group as the test data set. The remaining groups serves as the training data set. The model is then fitted using the training set and evaluated on the test set. The evaluation scores can then be retained, and the model can be discarded. Conclusively, the performance and quality of the model is summarised using the sample of the model evaluation scores. By validating the model with multiple train/test ratios and configurations it generally results in a less biased model [38].

3.4.3 Feature Selection

Feature selection is the idea to not include all features when training a model on a data set. By building more efficient and more comprehensible models, the risk of variance can be decreased, while interpretability and performance can be improved. Furthermore, by limiting the set of available features for the model, training time can be decreased [39]. In this study, careful feature analysis and evaluation was carried out not only to improve performance but also to further assess and evaluate the available features and their importance [40]. One method of measuring a features importance is by calculating its F-score and corresponding p-value (Equation 3.12). The F-score and corresponding p-value captures the linear relationship between features and labels. A highly correlated feature is given a higher F-score and the larger the F-score is, the more likely it is that this feature is more discriminative, as can be seen in Equation 3.12. Therefore, it can be suitable score as a feature selection criterion [41]. For every feature $X[:,i]$, the correlation C_i with target y is calculated and given this value, the F-score F_i and p-value p_i can be obtained through the equations:

$$\begin{gathered} C_i = \frac{(X[:,i] - mean(X[:,i])) * (y - mean(y))}{std(X[:,i]) * std(y)} \\ , F_i = \frac{C_i^2}{1 - C_i^2} * (n-1) \\ \text{and } p_i = 1 - \text{CDF}(F_i, 1, n-1) \end{gathered} \tag{3.12}$$

where n is the target sample size, std is the standard deviation and CDF is the cumulative distribution function for a given input $X = F_i$ and $d_1 = 1$ respectively $d_2 = n - 1$ degrees of freedom [42].

Another method to measure the significance of a feature before prediction is by estimating the mutual information of each feature. Mutual information between two variables measures the dependency between the variables. It is a non-negative value and is equal to zero if and only if two random variables are independent, a higher value means a higher dependency. The method of estimating mutual information presented in this study relies on nonparametric methods based on entropy estimation from k-nearest neighbours distances [43, 44].

3.5 Performance Metrics for Regression Methods

Evaluating MLAs requires metrics in order to assess their predictions and to compare them against each other. A common error metric for measuring the goodness of a fit of a model is the R^2 score [16, 20]. It compares the true values to the predicted value of the model and serves as measure of how well future samples are likely to be predicted. The R^2 score has both its weaknesses and strengths but can in combination with an error metric serve as purposeful metric [45, 46]. Another common method is to measure the absolute error of predicted values against true values. This is seen in multiple studies in the related area where the most common absolute error metric is MAPE [9, 16, 17, 21]. One of the issues with MAPE is that since it measures the absolute percentage one must divide the absolute error with the actual value. In a meeting with the airline company it was concluded that there is a possibility of the actual values being zero which makes MAPE ineffective and meaningless [47]. Therefore, other error metrics had to be evaluated and selected. In this study, Mean Absolute Error (MAE) is used as an error metric as research has proven it to be a superior error metric compared to others [48]. Furthermore, in many MLAs the Mean Squared Error (MSE) is used when creating the model and for cross-validation. MSE is widely used and has proven to be of value as an error metric in previous studies in related fields [49, 50, 51].

In conclusion, the metrics used in this study for evaluating and comparing MLAs are R^2 score, MSE and MAE. These metrics are used both when constructing the models and when comparing them against one another.

3.5.1 Mean Squared Error

The MSE measures the average of squared errors and is used in this study as a tool for comparing models and for cross-validation. In cross-validation a portion of the training data for the model is held back and not used when estimating the model. The MSE is evaluated on this portion and is then often referred to as mean squared prediction error (MSPE). The MSE and MSPE estimated over n samples and the portion q is defined as

$$\begin{aligned} \text{MSE}(y,\hat{y}) &= \frac{1}{n}\sum_{i=0}^{n-1}(y_i-\hat{y}_i)^2 \\ \text{and MSPE}(y,\hat{y}) &= \frac{1}{q}\sum_{i=n+1}^{n+q}(y_i-\hat{y}_i)^2 \end{aligned} \tag{3.13}$$

where $\hat{y}_i$ is the models predicted value of the i-th sample and y_i is the corresponding true value [52].

3.5.2 Mean Absolute Error

The MAE serves as unambiguous average-error magnitude for models and has proven to be a more natural measure of error compared to other average error measures such as root-mean-square error (RMSE). The MAE estimated over n samples is defined as

$$\text{MAE}(y,\hat{y}) = \frac{1}{n}\sum_{i=0}^{n-1}|y_i-\hat{y}_i| \tag{3.14}$$

where $\hat{y}_i$ is the models predicted value of the i-th sample and y_i is the corresponding true value [48].

3.5.3 Coefficient of Determination

The R^2 score is a measure of how good a fit the prediction is compared to the true values. The score serves as a measure of how well future samples are likely to be predicted by the model, based on the proportion of total variation of outcomes explained by the model. The highest achievable score is 1 and there is no lower boundary as the model can be arbitrarily worse. A model which disregards the input features and constantly predict the expected value would get an R^2 score of 0.

The R^2 score estimated over n samples is defined as

$$\begin{aligned} R^2(y,\hat{y}) &= 1-\frac{\sum_{i=0}^{n-1}(y_i-\hat{y}_i)^2}{\sum_{i=0}^{n-1}(y_i-\bar{y})^2} \\ \text{where } \bar{y} &= \frac{1}{n}\sum_{i=0}^{n-1}y_i \end{aligned} \tag{3.15}$$

and $\hat{y}_i$ is the models predicted value of the i-th sample and y_i is the corresponding true value [46, 45].

3.6 Statistical Analysis for Distribution Comparison

In order to assess if there is any statistical significant difference between two or more distributions, statistical tests are a requirement [53]. In this study, the three more advanced evaluated MLAs, namely SVR, XGBoost and MLP, are compared separately towards the LR baseline. For these statistical tests on comparing one MLA against the LR baseline, the nonparametric goodness-of-fit statistical Kolmogorov-Smirnov test is applied [54]. In a study by S Cankurt and A Subasi on creating customer demand models using a MLP model and a SVR model, the Kolmogorov-Smirnov test is applied for detecting if there is any significant evidence for proving seasonal patterns in the obtained sample [55]. Conclusively, in a study by Rand R Wilcox, the Kolmogorov-Smirnov test is proven to be a competitive goodness-of-fit statistical analysis test in regard to other comparable tests [56]. These studies demonstrates the applicability of the selected Kolmogorov-Smirnov test and serves as a basis for the selection of a nonparametric goodness-of-fit statistical test for comparing two distributions.

Furthermore, to verify if there is a significant difference between the three more advanced models in regard to each other, Kruskal-Wallis statistical tests are applied. A Kruskal-Wallis test is a nonparametric statistic for comparing three or more independently sampled groups on a single, non-normally distributed continuous variable [57]. The test identifies if there is a statistical difference between the samples but is unable to identify where the difference occurs [53]. The Kruskal-Wallis test is selected due to its ability of comparing three or more samples and because of the tests applicability and results when comparing samples in related customer demand and sales forecasting studies [58, 59] and in studies comparing multiple statistical tests [60, 61].

3.6.1 The Kolmogorov-Smirnov Test

The Kolmogorov-Smirnov statistic value D given two samples is calculated by computing the observed cumulative distribution functions of the two samples and then computing their maximum difference. The test is used to test the null hypothesis *H0 : the samples from X come from Y* against *H1 : the samples from X do not come from Y*, where the null hypothesis is rejected if $D > D_\alpha$. D is the calculated value and D_α is given by the Kolmogorov-Smirnov table with a significance level α and a number of samples in each group[54].

3.6.2 The Kruskal-Wallis Test

For the Kruskal-Wallis one-way analysis of variance test, the null hypothesis is that the given samples are all drawn from the same distribution, *H0: All sample distributions are equal*. A rejection of the null hypothesis indicates that one or more samples dominate another sample *H1: One or more sample distributions are not equal*. The null hypothesis is rejected if $H > H_\alpha$, where H is the calculated value and H_α is the table value. The table value is calculated with a given α and the number of degrees of freedom, which is $k-1$, where k is the number of groups. The value H is calculated as Equation 3.16.

$$H = \frac{12}{N(N+1)} \sum_{j=1}^{k} \frac{R_j^2}{n_j} - 3(N+1) \tag{3.16}$$

where N is the total sample size of all groups, n_j is the sample size of the jth group and R_j is the sum of the ranks in the jth group [53].

Chapter 4

Method

In this chapter, the chosen method and data set used for the experiments are described. The process of preparing the data set for the four selected MLAs is split into three steps, cleaning, feature engineering and final preparations. These steps are thoroughly outlined and explained. Furthermore, the implementation, design choices and evaluation method of the four selected MLAs are illustrated and described.

4.1 Material

In this section, the material examined in this study is thoroughly explained. The data solely consist of a single airline's flights and in order to cohere with the limitations presented in this study, various filters was applied on the data set. The data originates from the airlines in-house database and was collected as a file dump from the airline's representative March 8th, 2019.

4.1.1 Structure and Scope of Material

The data is structured as a matrix where each row represent a flight. Each flight consists of data specific to this flight and all features included in the original raw data set are presented in Appendix A. The data covers historic flights spanning a three-year period up until March 2019 with a total 850,887 entries with 58 features. The data set covers all flights in eight route areas whereas six of these are flights with in-flight meals for sale. These six route areas consists of flights up to four hours long and covers mostly flights in the European area. The final features selected post preprocessing for the MLAs are described in the following section, specifically in Table 4.1 and Table 4.2.

The features in Appendix A starting with "*No of sold/served...*" are the features containing information about the number of sold meals for each food category. A total of 9 such categories exists, and these are further referred to as "*Sold food categories*". The last 35 features presented in Appendix A are different special food category. These features consist of four capital letters and contain information about the number of loaded meals of each special food category and are referred to as "*Special meal categories*". These features were deemed as necessary as they are a requirement for the engineering of the target label (Section 4.2.2).

Historical data on load level of in-flight meals, where load level refers to the actual amount of meals loaded for each and every flight, was not in the data set. It was deemed confidential [47] and has as such not been used in any calculation, MLA or comparison. It was not a requirement for the predictions as the predictions of the MLAs are on customer demand of in-flight meals and not on predicting the current means of estimation. The current means of estimation could however prove useful when assessing the MLAs applicability to the problem and predictions. This evaluation was not considered for this study due to the confidentiality of the data and as the study aimed to evaluated MLAs against each other.

4.2 Data Preprocessing

Data preprocessing is the process of preparing the raw data set for the MLA models. The data preprocessing was split in three steps. The first step, cleaning, was to remove certain flights which were considered as unfit for the proposed method. The second part was feature engineering for the remaining flights in the data set. The third and final step was to prepare the data set for the models, this meant adding one hot encoders, normalisation and splitting into train and test sets. Multiple previous similar studies were taken in consideration as well as multiple meetings with the airline company when constructing and executing these three steps. Clustering of flights with similar attributes was considered as out of scope for this study.

4.2.1 Cleaning

Flights were removed according to five criteria. The first criteria concerned flights which were outside of the specified route area. This had to be done as the airline company does not conduct any sales of in-flight meals for two out of the eight route area occurring in the data set [47]. The second criteria were to remove flights within the specified route area but with no sales of in-flight meals. As the load level of in-flights meal was deemed confidential, all flights affected by this criteria was presented by the airline company [47] and could as such be removed from the data set. These entries almost entirely consisted of flights with a very short scheduled flight time. The third criteria were to remove flights where the amount of passengers ("*Final pax*") are zero as it would be impossible for sales to occur if there are no available customers. If these flights wouldn't be removed from the data set, the model would view the flight as a flight with zero sales which could affect the prediction. The fourth criteria concerns sales of breakfast meals during flights. In a meeting with the airline company [3] it was specified that the selling of breakfasts as an in-flight meal had recently concluded. This meant that in the historical data set there would be flights with data on breakfast sales. All flights concerning breakfasts were removed as a limitation on this study was to view meals as one feature. This means that the models would not be able to distinguish between different categories of in-flight meals.

The fifth and final criteria concerns the target label "*Sold load level*" (Equation 4.2) which represents a percentage where a value of 1 would result in a meal for each passenger. 52 flights were found with a "*Sold load level*" above 1 and were removed from the resulting data set as they were seen as anomalies [62]. Further anomaly detection revealed flights with identical arrival and departure stations and these occurrences were removed as well.

4.2.2 Feature Engineering

Feature engineering is the process of selecting and constructing the final features for model fitting and prediction. Feature selection is the method of disregarding features not suitable to be taken into consideration for the selected MLAs. This was done in collaboration with the airline company in order to ensure that the features selected for the MLAs are features available for future predictions. When the airline company requests a load level on the amount of in-flight meals for a specific flight, they alone possess the station pair, route area and the different features concerning time and date. In order for the models to be able to make future predictions, the features that were selected to be excluded were "*Final pax*", "*Sold food categories*", "*Special meal categories*", "*Flight number*" and "*Booking class*" [47]. No further feature selection was made on the remaining 10 features as they were all deemed necessary for the MLAs and further feature selection methods were considered as out of scope due to a limited time frame. Feature correlation and data analysis is presented in Chapter 5.

Using the flight history, seven continuous features were engineered for each flight (Table 4.1) and three categorical features (Table 4.2).

Table 4.1: Table of non-categorical features calculated for each flight.

Feature	**Mean**	**Standard Deviation**
Scheduled time of departure (STD)	-0.345730	0.573896
Scheduled time of arrival (STA)	-0.181904	0.676363
Weekday	3.782507	1.953430
Scheduled time	93.755483	43.324846
Day	15.711802	8.757178
Month	6.618611	3.350008
Year	17.164506	0.887054

Table 4.2: Table of categorical features calculated for each flight.

Feature	**Distinct count**
Departure station	125
Arrival station	124
Route area	6

The original features "*Date*" and "*yymm*" conveyed the same information and were converted into three separate features, "*Day*", "*Month*" and "*Year*" and the original features were disregarded. The feature "*Scheduled time*", which refers to the scheduled flight duration time, was originally in the format "hh:mm" and was reconfigured into minutes, a continuous value. The feature "*Weekday*" was already in a desired format of a number representing the weekday (1-7) and was as such not further engineered. The two features "*Scheduled time of departure*" and "*Scheduled time of arrival*" were both reconfigured from the original format "hhmm" (hour and minute of the day) into a continuous value on the range of -1 to 1 (see Equation 4.1). This was to implicate that the time of the day works as a circle where 23:59 is as closest to 00:00 and not the opposite.

$$\cos \frac{2 * \pi * \text{scheduled time in seconds}}{\text{seconds in a day (86400)}} \tag{4.1}$$

The categorical features selected for the MLA predictions are "*Departure station*", "*Arrival station*" and "*Route area*". These features were already in the data set in the desired format and was as such not further engineered.

Target Label

The target label was not initially a feature in the data set and had to be engineered, this was labelled "*Sold load level*" and is calculated as Equation 4.2. This was made in collaboration with the airline company as it is the desired format when discussing how many meals to load onto a flight [47]. "*Sold load level*" refers to the number of meals divided by number of possible customers. If a passenger has requested a special meal, they are not considered as a possible customer for the in-flight meal service as they have already purchased a meal. Therefore, the sum of the "*Special meal categories*" requires to be subtracted from the total amount of passengers ("*Final pax*").

$$\text{Sold load level} = \frac{\sum \text{Sold food categories}}{\text{Final pax} - \sum \text{Special meal categories}} \tag{4.2}$$

4.2.3 Preparation

The final part of the data preprocessing was to prepare the data for the model which was done in three parts. The first step was to field-wise one-hot encode the categorical features ("*Departure station*", "*Arrival station*" and "*Route area*"). For each feature, e.g., "*Departure station*", there are multiple units, each of which represents a specific value of this field, e.g., Departure station = ARN, and there is only one positive (1) unit, while all others are negative (0). Previous studies have confirmed the promising results of one hot encoding multiple categorical features for MLAs [63, 64].

The second step was to normalise the data. Multiple methods of normalisation exist and the selected method for this study was using a min-max-scaler [65] on the standard range of 0.0 to 1.0. The equation for the normalisation is shown in Equation 4.3.

$$\begin{aligned} y &= \text{std}(X) * (1.0 - 0.0) + 0.0 \\ \text{where std}(X) &= \frac{X - \min(X)}{\max(X) - \min(X)} \end{aligned} \tag{4.3}$$

The last part of the preparation was to split the data set into a train and test set respectively. Two methods were taken into consideration, either to select a subset at random or to select historically. After a meeting with the airline company [62], it was concluded to select a percentage based on time as the companies directives often change and as such the newest flights would most likely represent future cases best. As the data set consisted of three years of historical data, the last year was selected as the test set which represented approximately 33% of the total amount of available flights post preprocessing.

4.3 Execution

4.3.1 Data Analysis

Due to a limited time frame, the only data analysis is made post preprocessing and before model evaluation and prediction. This refers to explaining the features in the preprocessed data set in correlation with each other and towards the target values. This is done through a correlation heat map supplied from the Seaborn library [66] in the Python programming language and through invoking Equation 3.12 on all available features. In equation 3.12, a highly correlated feature is given a higher F-score and less correlated features are given a lower score. As F-score and p-value for evaluating the feature significance only captures linear relationships between features and labels, the mutual information of the features are evaluated additionally.

Feature significance in regard to mutual information is based on k-nearest neighbours' methodology and measures the dependency of one feature to another. This is done through the mutual_info_regression method in the Feature Selection class in the scikit-learn library for the Python programming language [67]. A value of $k = 3$ in reference to the k-nearest neighbours algorithm is the standard value proposed in the study by Brian C Ross [43]. However, due to limited computational memory and a large data set, a $k = 2$ was considered for this study.

4.3.2 Model Evaluation

The primary evaluation metrics are the MSE, MAE and R^2 score values of the selected algorithms. These are evaluated by fitting the models using the train data set and then applying the fitted models on the test data set. The predictions made by the fitted models are then compared to the actual values of the target label of the test data set. When fitting the model, the algorithm will split the training data set into a train and validation data set in order to create a better model and further minimise the risk of overfitting [10]. This is implemented through a k-fold cross validation algorithm. Furthermore, the computational performance is also measured during both fitting of the model and during prediction of the test set of the model. This is also used as a metric when comparing the selected algorithms as it can be of vital importance if the algorithm has to handle many requests from a large airline company with a large amount of daily flights.

In this study, a baseline is presented for further assessment of the applicability of the results. A baseline is often a simpler or a more basic algorithm. As such, the predictions of the LR algorithm is selected to be evaluated as a baseline compared to the more advanced MLAs. The predictions of the LR models will serve as a basis for comparison.

To visualise the performance of the models, two graphs are used. One graph to directly compare the predicted values versus the true values and one graph for visualising the residuals. A residual is the difference between the observed value of the dependent variable and the predicted value, each data point has a residual. As such, a residual is calculated with the formula: Residual $=$ Observed value $-$ Predicted value. Residuals are used in many procedures and are designed to detect disagreements between the data and an assumed model. Model improvement possibilities can be detected by analysing if the residuals has any clear pattern or trend. Residual analysis can be and has been used to produce stronger and more compelling conclusions [68]. The regression analysis between the residuals and the predicted value was done by using a regression-smoothing technique called Locally Weighted

Scatterplot Smoothing (LOWESS), to highlight trends and patterns in the model predictions [69]. The fitted LOWESS is represented as line in the residual plot for each MLA.

Statistical Model Analysis

For analysing if there is a statistical differnece in terms of prediction accuracy between the models, Kolmogorov-Smirnov tests and Kruskal-Wallis tests are performed on the computed prediction accuracy metrics of the four evaluated MLAs. The tests are calculated as described in Section 3.6.1 and Section 3.6.2 respectively and are applied on the prediction accuracy metrics.

The Kolmogorov-Smirnov tests are executed with the LR as a baseline and in comparison to the other three more advanced MLAs, which results in a total of three tests.

1. The LR model against the SVR model.
2. The LR model against the XGBoost model.
3. The LR model against the MLP model.

The Kruskal-Wallis tests are executed to indicate if there is any significant difference between the more advanced models and between all models, which results in a total of two tests.

1. The SVR, XGBoost and the MLP models against each other.
2. The LR, SVR, XGBoost and the MLP models against each other.

The tests are computed to result in a $p\text{-}value$, with which we reject the null hypothesis if $p\text{-}value > \alpha$. All presented tests are executed with a significance level α of 0.05, as it is the typical value value presented in the literature for both the Kolmogorov Smirnov tests by Hodges and in the literature for Kruskal-Wallis tests by Corder and Foreman [54, 53]. Furthermore, due to a limited time frame, no other α values were considered for this study.

4.3.3 Model Implementation

The machine learning methods and data processing part of the study was carried out through the Python programming language using the StatsModels library, the scikit-learn library, the xgboost library and the Keras library through TensorFlow. In order to find the optimal parameters for each MLA, a grid search with k-fold cross-validation was implemented with a $k = 5$ where each fold was measured using MSE. There is no formal rule as to the value of k but it is commonly 5 or 10. These values have been shown empirically to yield test error rates estimates that suffer neither from excessively high bias nor from very high variance [38, 70]. Due to a limited time frame, a k value of 5 was selected.

The hyperparameter grid search consists of specifying a grid of hyperparameters with one or more corresponding possible value. The grid search algorithm creates a model with each set up of hyperparameter and uses a k-fold cross validation to assess the performance of the model. All possible set ups of the specified grid are created $k = 5$ times and evaluated against each other.

Linear Regression

The LR algorithm was implemented through the OLS (ordinary least squares) class in the Linear Regression module in the StatsModels library for Python. The algorithm calculates ordinary least squares Linear Regression [71]. As such, there was no hyperparameter grid search evaluated for the LR algorithm except for the k-fold cross validation. As such, the total number of models created and evaluated are 5.

Support Vector Regression

The SVR algorithm was implemented through the SVR class in the scikit-learn library for Python. The algorithm calculates epsilon-Support Vector Regression with a specified kernel, C and epsilon. Where C is the penalty parameter for the error term [72]. The implementation is based on libsvm [73]. Due to a limited time frame and long training times for an SVR model [21], limitations on the training data had to be made when finding the optimal hyperparameters. This has been seen in similar studies [6] and in order to make the search for optimal hyperparameters more efficient a randomly selected subset from the original training set was chosen. In Almérs study a subset of 10% is reasoned to be sufficient for a hyperparameter search and as such, 15% of the original data set is selected for the subset in this study. There is always the risk of a heavy bias when selecting a small subset for training and selection of hyperparameters [10] and as such the finalised model will further be compared towards a model with recommended hyperparameters. These recommended hyperparameters, kernel, C and ϵ, were selected for their results in similar studies [22, 73, 74] and are shown in Table 4.4. The evaluated values for the hyperparameters are shown in Table 4.3. For the grid search on the subset, a total of 2800 models are created and evaluated. Furthermore, on the complete data set, 5 models with the recommended hyperparameters and 5 models with the optimal hyperparameters from the grid search are created and evaluated.

Table 4.3: Evaluated hyperparameters for proposed SVR model.

Hyperparameter	**Possible values**
Kernel	linear, polynomial (2, 3), RBF, sigmoid
C	0.1, 0.2, 0.5, 0.9, 1, 1.25, 1.5, 2, 2.5, 3, 5, 7, 10, 15
Epsilon	0, 0.005, 0.01, 0.1, 0.2, 0.5, 1, 2

Table 4.4: Recommended hyperparameters for proposed SVR model.

Hyperparameter	**Value**
Kernel	RBF
C	1
Epsilon	0.1

Extreme Gradient Boosting

The XGBoost algorithm was implemented through the XGBRegresser class in the xgboost library for Python. The implementation and algorithm is based on the content from Tianqi

Chen [11, 75]. The hyperparameters selected for evaluation are limited due to a limited time frame and the span is configured to cover most cases (Table 4.5).

In any tree ensemble model the two most significant parameters are the depth of the trees and the number of trees [11, 76] and are as such evaluated towards a span of values in this study. Furthermore, in order to further prevent overfitting [11], the subsample ratio of columns when constructing each tree is also evaluated in this study. On the XGBoost algorithm, a total of 450 models are created and evaluated.

Table 4.5: Evaluated hyperparameters for proposed XGBoost model.

Hyperparameter	**Possible values**
Max depth	5, 6, 10, 15, 20, 25
Subsample ratio of columns when constructing each tree	0.5, 0.6, 0.7, 0.8, 0.9
Number of estimators	100, 200, 300

Multilayer Perceptron Neural Network

The MLP was implemented with the Keras Sequential model API through the TensorFlow package for Python [77, 78]. The number of hidden layers for the MLP was only considered to be one, as previous experiments and literature have proven there to be no significant improvement of the prediction's quality if more layers were to be added [37]. The remaining hyperparameters up for evaluation are shown in Table 4.6 where the span of possible values are configured to cover most cases whilst the activation function, optimiser and initialiser are limited due to a limited time frame. The selection of hyperparameters for the grid search and their possible values are selected from related studies for their promising results [6, 10, 17]. In order to estimate a continuous value, a linear activation function is selected for the output node. A total of 1260 models are created and evaluated for the MLP algorithm.

Table 4.6: Evaluated hyperparameters for proposed MLP model.

Hyperparameter	**Possible values**
Number of hidden layers	1
Activation function	ReLU, sigmoid, tanh
Number of hidden nodes	5, 10, 15, 20, 30, 50, 100
Optimiser	adam, rmsprop
Loss metric	MSE
Batch size	32, 64, 256
Initialiser	Glorot uniform
Number of epochs	100, 1000

Chapter 5

Result

The results of the experiments are presented in this chapter. First, the result of the data preprocessing is presented. Then, the result of the four MLAs predictions of customer demand of in-flight meals is shown. Finally, a section on data analysis and feature importance.

5.1 Extent of the Material

The historical data covered a period of three years of flights from a single airline company. Due to many anomalies and unpreprocessed entries in the data, a substantial amount of flights was removed during the preprocessing process. From the original 850,887 flights, 584,033 (68.64%) remained post preprocessing. These 69% were then split into training and test set respectively and configured with every MLA constructed. The training data set consisted of the first two thirds of the preprocessed data set whilst the test data set consisted of the last third, which represented the third and last year of entries. The preprocessing process of the original data set took 126s.

Table 5.1 explains the preprocessing process in terms of loss in percentage of each route area occurring in the original data set. All flights in the data operates inside one of these route areas.

Table 5.1: Percentage of each route area.

Route area	**% of original data set**	**% of processed data set**	**% of original area set**
1E	19	23.75	99.17
2D	25.87	12.02	36.85
2E	7.81	9.8	99.59
3D	14.02	17.23	97.62
3E	14.77	18.46	97.62
5I	15.21	18.74	97.74
0U	2.41	0	-
0A	0.91	0	-

5.2 Performance of Algorithms

This section provides the result of the k-fold cross validation of the selection of hyperparameters for the proposed MLAs. It includes results of the performance metrics on the predictions of said models with optimal hyperparameters. Four different MLAs (LR, SVR, XGBoost, MLP ANN) were evaluated and a total of 4,525 models were created and evaluated on the preprocessed data set. For each evaluated MLAs, a figure showing the predicted values compared to the true/observed values of the target "*Sold load level*" is included. A line representing a perfect predicted versus true/observed value is visualised in each figure. Furthermore, a residual plot for each MLAs is included. These figures are included to visualise the performance of the models in terms of prediction accuracy and for visualisation of any possible trends and patterns in the model predictions.

Table 5.2: Metric values for evaluated MLAs.

Metric	**LR**	**SVR**	**XGBoost**	**MLP**
MAE	0.0258	0.024	0.0239	0.0242
MSE	0.0020	0.0019	0.0017	0.0018
R^2 score	0.4464	0.4694	0.5124	0.4995
Model fit time (seconds)	32	308121	321	295
Model predict time (seconds)	<0	15523	3	4

The values displayed in Table 5.2 presents the performance results of every MLA with optimal hyperparameters found in their respective 5-fold cross validation hyperparameter grid search. The values presented in the SVR column corresponds to the results of the SVR model constructed on the complete preprocessed data set with the optimal hyperparameter found in the grid search on the 15% subset of the data set.

5.2.1 Linear Regression

Performance result of the LR MLA can be seen in Table 5.2. As the LR model calculated ordinary least squares Linear Regression, no hyperparameters were evaluated. A total of 5 models were created and evaluated.

The LR model predictions versus the true values of the test data set are visualised in Figure 5.1 with a corresponding line representing a perfect one to one relationship. Figure 5.2 shows the residual plot of the LR model with a corresponding fitted LOWESS line.

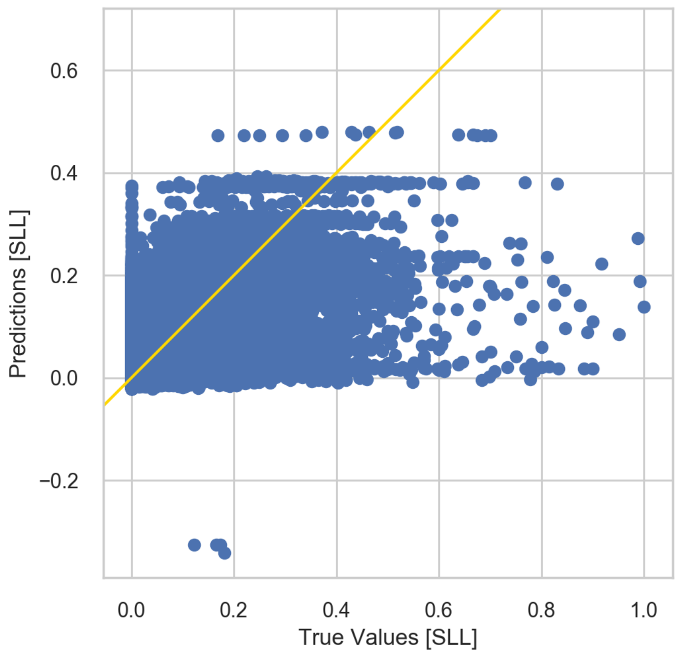


Figure 5.1: Predicted "*Sold load level*" (SLL) values of optimal LR model versus true/observed values of SLL on test data set. The dots are data points in the test data set and the line represents a perfect one to one relationship.

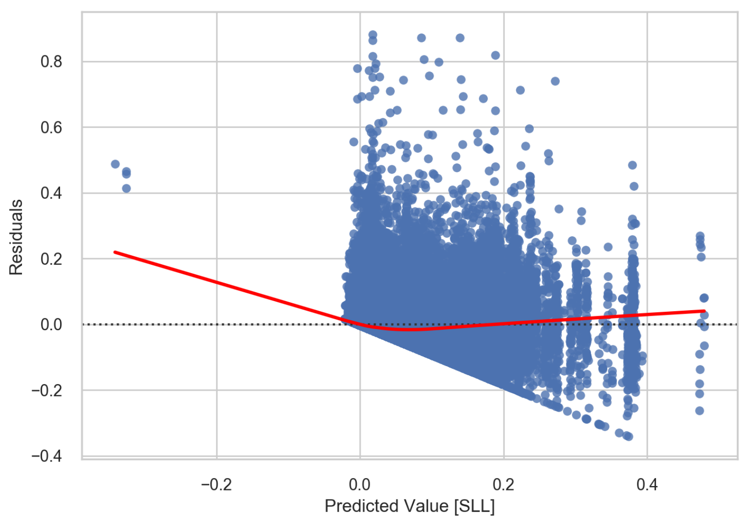


Figure 5.2: Residuals plot of optimal LR model predictions. The y-axis represents the residuals (Observed value − Predicted value) and the x-axis the predicted value of the target label "*Sold load level*" (SLL). The dots are data points in the test data set and the line is a fitted LOWESS line. The line is included to highlight trends and patterns in the model predictions.

5.2.2 Support Vector Regression

Optimal hyperparameters found in the 5-fold cross validation hyperparameter grid search on the 15% subset along with the recommended set up of hyperparameters of the SVR MLA can be seen in Table 5.3. The results of both recommended and optimal hyperparameters are displayed in Table 5.4. In these tables, "Recommended" refers to the hyperparameters and performance of the model with the recommended hyperparameters. "Optimal" refers to the hyperparameter found in the grid search on a subset of 15% of the preprocessed training and test data set. For the subset, a total of 2800 models were created and evaluated and 10 models for the whole data set.

Table 5.3: Hyperparameters for proposed SVR model.

Hyperparameter	Recommended Value	Optimal Value
Kernel	RBF	RBF
C	1	10
Epsilon	0.1	0.01

Table 5.4: Metric values for proposed SVR models. Rec (Recommended) refers to the recommended settings of hyperparameters and Opt (Optimal) to the parameters found in the grid search. The percentage is the amount of preprocessed data that was used in training and testing the model.

Metric	Value (Rec/100%)	Value (Opt/15%)	Value (Opt/100%)
MAE	0.0685	0.0247	0.024
MSE	0.0058	0.0020	0.0019
R2 score	-0.5776	0.4714	0.4694
Model fit time	213411 seconds	775 seconds	308121 seconds
Model predict time	97 seconds	7 seconds	15523 seconds

The result of the SVR model with recommended setting and with the optimal settings found in the grid search on both the subset and on the whole data are displayed in Table 5.4. The recommended settings underperform in all performance metrics concerning prediction accuracy but is an improvement compared to the optimal settings on the whole data set in terms of model fit and prediction time. The resulting MAE, MSE and R^2 score values from the SVR model with optimal hyperparameters on the 15% subset are close to the results of the SVR model with the same hyperparameters on the whole data set, in regard to model prediction accuracy.

The SVR model predictions versus the true values of the test data set are visualised in Figure 5.3 with a corresponding line representing a perfect one to one relationship. Figure 5.4 shows the residual plot of the SVR model with a corresponding fitted LOWESS line.

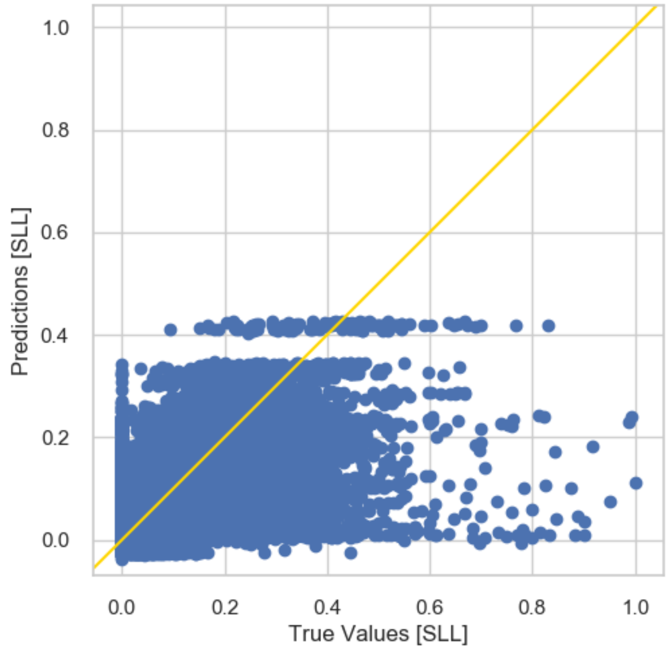


Figure 5.3: Predicted "*Sold load level*" (SLL) values of optimal SVR model versus true/observed values of SLL on test data set. The dots are data points in the test data set and the line represents a perfect one to one relationship.

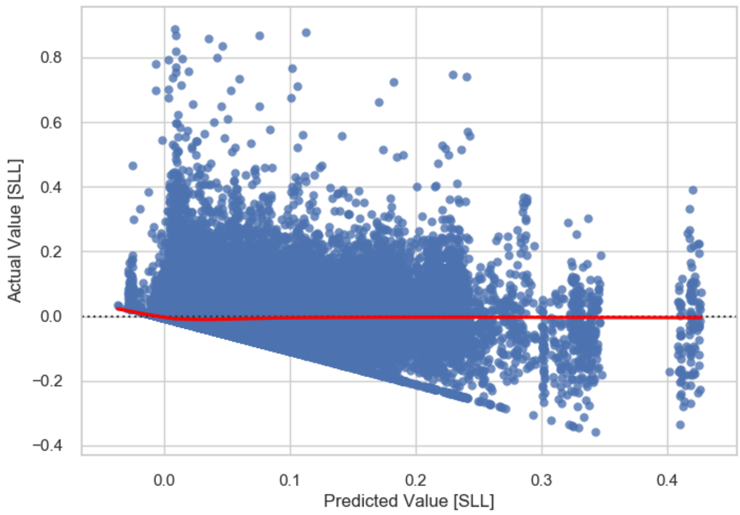


Figure 5.4: Residuals plot of optimal SVR model predictions. The y-axis represents the residuals (Observed value − Predicted value) and the x-axis the predicted value of the target label "*Sold load level*" (SLL). The dots are data points in the test data set and the line is a fitted LOWESS line. The line is included to highlight trends and patterns in the model predictions.

5.2.3 Extreme Gradient Boosting

Optimal hyperparameters found in the 5-fold cross validation hyperparameter grid search can be seen in Table 5.5. The performance results of the XGBoost model with optimal hyperparameters found in the grid search are shown in Table 5.2. For the grid search, a total of 450 models were created and evaluated.

Table 5.5: Hyperparameters for proposed XGBoost model.

Hyperparameter	**Optimal value**
Max depth	10
Subsample ratio of columns when constructing each tree	0.5
Number of estimators	100

The XGBoost model predictions versus the true values of the test data set are visualised in Figure 5.5 with a corresponding line representing a perfect one to one relationship. Figure 5.6 shows the residual plot of the XGBoost model with a corresponding fitted LOWESS line.

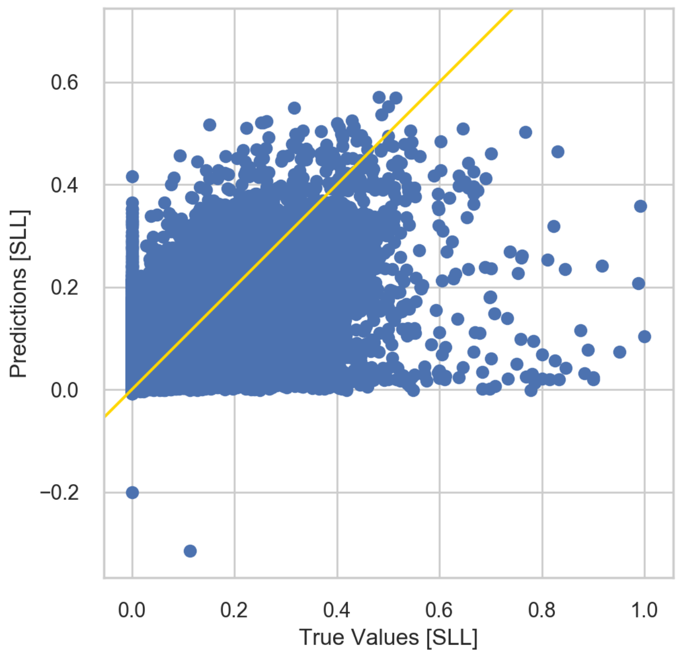


Figure 5.5: Predicted "*Sold load level*" (SLL) values of optimal XGBoost model versus true/observed values of SLL on test data set. The dots are data points in the test data set and the line represents a perfect one to one relationship.

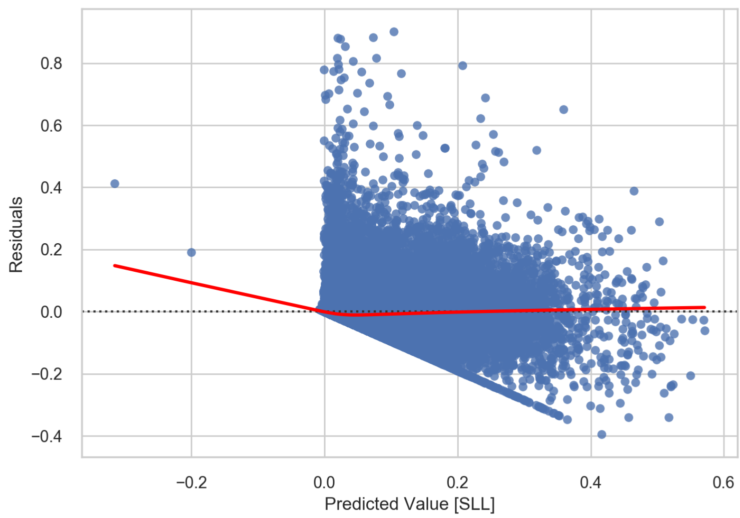


Figure 5.6: Residuals plot of optimal XGBoost model predictions. The y-axis represents the residuals (Observed value − Predicted value) and the x-axis the predicted value of the target label "*Sold load level*" (SLL). The dots are data points in the test data set and the line is a fitted LOWESS line. The line is included to highlight trends and patterns in the model predictions.

5.2.4 Multilayer Perceptron Neural Network

Optimal hyperparameters found in the 5-fold cross validation hyperparameter grid search can be seen in Table 5.6. The performance results of the MLP model with optimal hyperparameters found in the grid search are shown in Table 5.2. For the grid search, a total of 1260 models were created and evaluated.

Table 5.6: Evaluated hyperparameters for proposed MLP model.

Hyperparameter	**Optimal value**
Number of hidden layers	1
Activation function	ReLU
Number of hidden nodes	50
Optimiser	adam
Loss metric	MSE
Batch size	32
Initialiser	Glorot uniform
Number of epochs	1000

The MLP model predictions versus the true values of the test data set are visualised in Figure 5.7 with a corresponding line representing a perfect one to one relationship. Figure 5.8 shows the residual plot of the MLP model with a corresponding fitted LOWESS line.

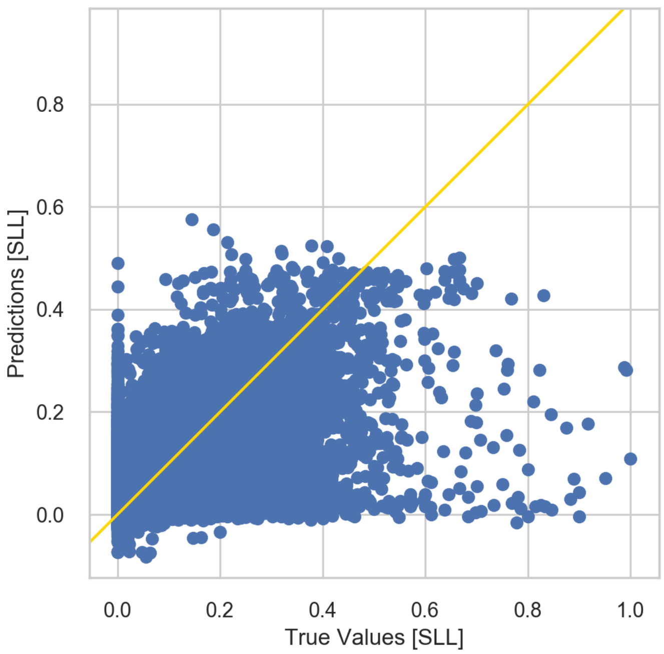


Figure 5.7: Predicted "*Sold load level*" (SLL) values of optimal MLP model versus true/observed values of SLL on test data set. The dots are data points in the test data set and the line represents a perfect one to one relationship.

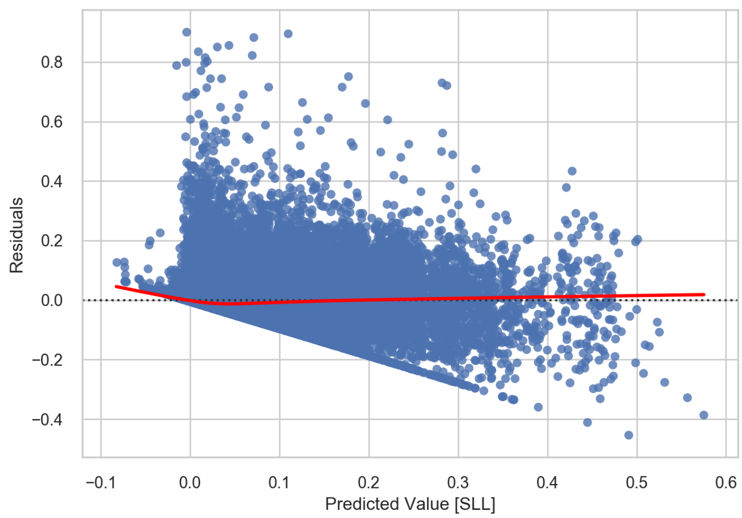


Figure 5.8: Residuals plot of optimal MLP model predictions. The y-axis represents the residuals (Observed value − Predicted value) and the x-axis the predicted value of the target label "*Sold load level*" (SLL). The dots are data points in the test data set and the line is a fitted LOWESS line. The line is included to highlight trends and patterns in the model predictions.

5.3 Statistical Analysis

All values presented in this section are rounded to two decimal points. A total of three Kolmogorov-Smirnov tests were executed on the resulting model accuracy prediction metrics (MSE, MAE and R^2 score), with an α-value of $\alpha = 0.05$.

1. Kolmogorov-Smirnov test on the LR model against the SVR model.
   Result: $p\text{-}value = 0.98$

2. Kolmogorov-Smirnov test on the LR model against the XGBoost model.
   Result: $p\text{-}value = 0.98$

3. Kolmogorov-Smirnov test on the LR model against the MLP model.
   Result: $p\text{-}value = 0.98$

A total of two Kruskal-Wallis tests were executed on the resulting model accuracy prediction metrics (MSE, MAE and R^2 score), with an α-value of $\alpha = 0.05$.

1. Kruskal Wallis-test on the SVR, XGBoost and the MLP models.
   Result: $p\text{-}value = 0.96$

2. Kruskal Wallis-test on the LR, SVR, XGBoost and the MLP models.
   Result: $p\text{-}value = 0.99$

As the $p\text{-}value$ is high and larger than the α-value we cannot reject the null hypothesis for all five presented tests. Concluding in that for the Kolmogorov-Smirnov tests the three more advanced models, SVR, XGBoost and MLP perform with no statistically significant difference in comparison towards the LR baseline. Furthermore, the Kruskal-Wallis tests demonstrates that the four models prediction accuracy, in terms of the applied accuracy metrics, are not statistically significantly different.

5.4 Data Analysis

Due to the large number of 262 unique features in the available data set post preprocessing, only a subset of the features could be evaluated in a correlation matrix (Figure 5.9). The 7 continuous engineered features were selected and the 6 route areas where sales of in-flight meals were conducted. The remaining 249 features concerns the one hot encoded features "*Departure station*" and "*Arrival station*". As these engineered features represents ~95% of the possible features to select from, it was concluded to be sufficient to remove these from the feature correlation heat map.

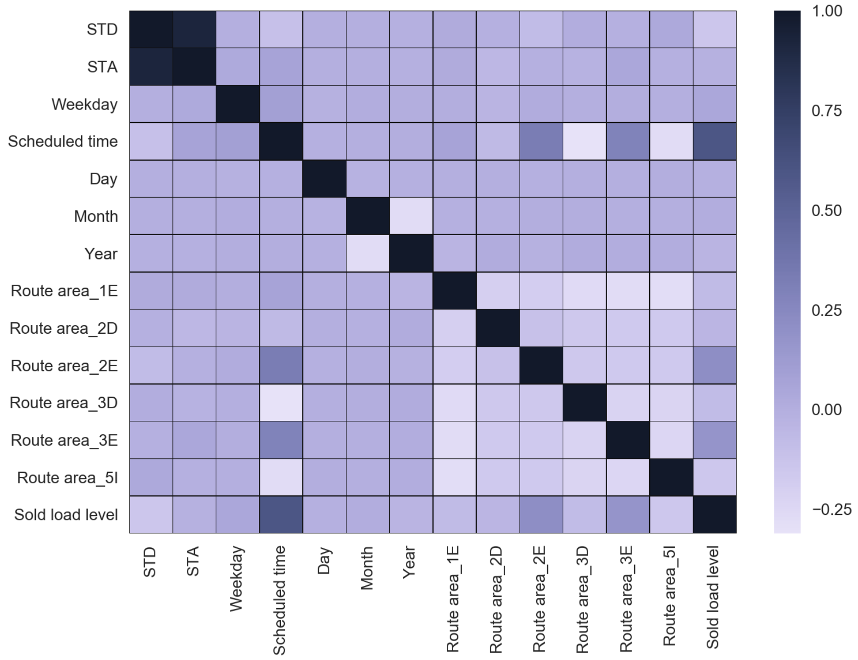


Figure 5.9: Correlation heat map of the 7 continuous features and the 6 route areas of the preprocessed data set.

The correlation heat map (Figure 5.9) shows a darker (purple) colour for the features which are found to be more correlated with each other. At the bottom and at the far right, the target label "*Sold load level*" is presented. The heat map shows that the feature "*Scheduled time*" has the highest correlation with the target label.

5.4.1 Feature Importance

For each feature, a F-score, a p-value and a mutual information value have been calculated and are presented in Table B.1 in Appendix B. The five highest scoring features in F-score and p-value are "*Scheduled time*", "*Route area_2E*", "*Arrival station_LPA*", "*Route area_3E*" and "*Departure station_AGP*". Respectively, the five highest scoring, regarding mutual information, are "*Scheduled time*", "*STD*", "*STA*", "*Route area_2E*" and "*Route area_5I*".

Chapter 6

Discussion

In this section, the thesis is discussed from several perspectives. The expected and unexpected consequences of working with a large data set are discussed in this chapter. The chosen method is reviewed and data analysis along with performance analysis are conducted and discussed. Finally, the ethical aspects are reviewed.

6.1 Material Limitations

The available data for this study consisted of three years of flight data originating from a single airline company, with initially close to a million entries. After data preprocessing, only 69% of the original data set remained, due to the filters applied. The filters consisted of removal of anomalies and of flights that were considered out of scope. The filtering process was consistent with the directives from the airline and the scope of the study. Much of the data was believed to have been filtered when removing flights with no sales of in-flight meals and flights selling in-flight breakfasts. The remaining data set post preprocessing consists of flights suitable for estimation and for future estimation. As such, the loss of 31% was not seen as a crucial factor towards the result but nevertheless a contributing one.

The set of features selected for the MLA models have to be available when specifying a load level for a future flight. As such, the selection process of said features was made in collaboration and in discussion with the airline company. As a consequence of this, there is a possibility that important features for the model predictions were removed in the preprocessing of the data. Features which could have contributed to the performance of the MLAs, such as the amount of passengers, are excluded from the models. Furthermore, if accessed had been gained to the current means of estimation and the historical load level of each flight in the data set, the applicability of the MLAs evaluated could have been more thoroughly investigated.

As the data set was large, performing the experiments of the four MLAs was a time consuming and compute intensive task. As such, the k-fold cross validation was limited to $k = 5$ and the grid search over possible hyperparameters for each MLA was further limited. Further limitations were on the SVR MLA as the grid search was only evaluated on a subset of 15% of the preprocessed data set. As this could achieve a heavy biased model, a larger span of possible hyperparameters for this grid search was evaluated and a recommended set up of hyperparameters for the SVR were further analysed and compared.

The reason for these limitations was to achieve a reasonable execution time when the different experiments were performed, not to create a state-of-the-art application. Without the limitations, the experiments were considered unfeasible due to the time frame of this study. Therefore, the limitations and optimisations might not have been optimal.

6.1.1 Discarded Material and Rogue Data

The filters created for discarding rogue data were constructed in collaboration and in discussion with the airline company. As such, the only anomalies detected was during meetings with the company and further data analysis when constructing the preprocessing process. Therefore, the possibility of further anomalies still included in the remaining 69% exists. In order to assess their importance and influence over the result, further data analysis has to be performed. Due to a limited time frame, it was considered as out of scope for this study. Furthermore, as the airline company specified that breakfasts was no longer for sale as an in-flight meal these had to be removed as well. This was done by removing all flights which had recorded a sale of a breakfast in-flight meal. This could have the effect of flights with sales of both breakfasts and another food category being removed and flights with loaded breakfasts but no sales of this category getting to stay in the final data set. As a complete set of criteria for identifying these flights could not be obtained through the airline company, this method of removal was seen as sufficient for this study.

Table 5.1 explains the loss of entries in relation to the route area in which they are operating inside. The route areas “0U” and “0A” are completely removed from the data set, as no sales of in-flight meals are conducted on these flights. The distribution of the sizes of the route areas occurring in the preprocessed data set is more balanced than before preprocessing and the route area “2D” has lost more than half of its flights. This is assumed to be because of many flights serving breakfasts and many flights not serving any meals are operating inside that route area.

6.2 Method Analysis

All four MLAs were evaluated and selected in the same manner of a hyperparameter grid search, with the difference that the SVR implementation used a subset for the grid search. The span of possible values for the hyperparameters was configured to cover as many cases as possible, while not being too time consuming. This meant that for some MLAs, more models of different configuration schemes were evaluated compared to other MLAs which could have influenced the results. Furthermore, for the MLP MLA, a more suitable approach of finding optimal hyperparameters could be to implement early stopping and model checkpoints in order to find the most optimal model. This was seen as out of scope for this study due to a limited time frame. However, the value spans presented in the grid search were all surrounding recommended value spans presented by the MLAs original authors and/or studies and literature in the field. As such, the hyperparameter grid search was seen as sufficient and further hyperparameter grid search with a wider span could prove to be unnecessary. As for the method of selecting hyperparameters for the SVR model, Table 5.4 presents the resulting performance metrics of said model on the test data set. The SVR model with optimal hyperparameters on the 15% subset (Column “Value (Opt/15%)”) has similar results to the SVR model with optimal hyperparameters on the whole data set (Column “Value (Opt/100%)”), proving the applicability of the selected method of finding optimal hyperparameters for an SVR model.

Further method limitations due to a limited time frame was made on the normalisation technique. The only technique considered was using a min-max-scaler, due to its results in previous studies. Further analysis of different normalisation techniques and different spans of normalisation values could further improve the results of the MLAs considered. Evaluating more MLAs was considered as out of scope for this study, partly due to the limited time frame but mainly as a result of the study on related previous research. The conclusions obtained from the study on related work demonstrated promising results in evaluating the selected MLAs for predicting the customer demand of in-flight meals. Accordingly, evaluating more MLAs was considered unnecessary and in order to further improve the results obtained, focus should primarily be put towards further optimisation of the selected MLAs for this study and data preprocessing and feature engineering techniques.

6.3 Data analysis

Feature selection was not considered for this study apart from feature significance analysis due to a limited time frame. Selecting a subset of features could further improve the results obtained and decrease the model fit and prediction time. The feature significance analysis results in three evaluated metrics for all features available post preprocessing (Appendix B) and was run on an Amazon Web Services (AWS) m5.24xlarge Linux/UNIX Spot Instance.

The three most contributing and important features for predicting customer demand of in-flight meals, in terms of the mutual information value based on k-nearest neighbour with a $k = 2$, are the flight duration time and the time of departure and arrival. This metric was seen as the most comprehensive as it is capable of capturing nonlinear dependencies and the dependency of one feature against another. Furthermore, the feature "*Scheduled time*", which refers to the flight duration time, has a substantially larger value in F-score, p-value and mutual information value than all other features. This proves, in regard to the metrics presented, that the flight duration time has the largest impact on the customer demand of in-flight meals. Accordingly, this can be seen in the correlation heat map presented in the results (Figure 5.9). In the figure, the target label "*Sold load level*" has the highest correlation value with the "*Scheduled time*" feature.

Many of the features available for the evaluated MLA models separately describes the same information. The feature "*Scheduled time*" is indirectly included in the features "*STD*", "*STA*" and the airport city pair, the departure and arrival station. Furthermore, the city pair also contain the information of route area. Including many features containing the same information, as was done in this study, could influence the model performance results, for better or for worse. In order to assess the importance and significance of the features, further data analysis, feature significance analysis and feature selection have to conducted.

6.4 Analysis of Performance Results

The proposed models of the four MLAs have all been selected through evaluating multiple models where the internal evaluations were made based on MSE. When comparing and evaluating the models however, multiple metrics were considered, namely MSE, MAE, R^2 score and model fit and prediction time. This results in that the final models selected through the hyperparameter grid search are compared against metrics that the models were not selected for during the search. As such, different hyperparameter configuration schemes of the models could have a better MAE and R^2 score than the resulting models.

6.4.1 Hyperparameter Search

The hyperparameter grid search was run on an AWS m5.24xlarge Linux/UNIX Spot Instance. The grid searches for the four MLAs took up to 72 hours each to complete. The LR models were fastest and then came the XGBoost models, the SVR models and finally the most time consuming one, the MLP models. A wider grid search on each MLA with more hyperparameters and a larger span could potentially improve the models and the model's predictions, pinpointing it further to predicting customer demand of in-flight meals. Because of the limited time frame, the span values are configured to cover cases surrounding recommended values of the different MLAs.

6.4.2 Performance Metrics and Figures

To analyse the performance result of the LR model, the resulting figures (Figures 5.1 and 5.2) and the performance metrics results (Table 5.2) are considered and discussed. The LR model was used as a baseline for comparison against the three more advanced MLAs (SVR, XGBoost, MLP) and the results in regard to performance accuracy are marginally worse than the other MLAs. Due to the simplicity of the model, it is the top performing MLA regarding model fit and predict time. The drawback of the LR model is that we have no further hyperparameters available for evaluation, which means that apart from further data preprocessing and feature selection, the results of the LR model are as good as they get. Compared to the performance metrics of this LR baseline, all remaining three MLAs performs marginally better in terms of prediction accuracy and are to an extent capable of identifying the customer demand of in-flight meals. Out of the two proposed SVR models presented, the one configured used the optimal hyperparameter values found in the grid search on the subset outperforms the proposed SVR model configured using the recommended hyperparameters in terms of the performance metrics regarding model prediction accuracy. Therefore, it is this one that is referred to when discussing the proposed SVR model henceforth.

By looking at the figures for the proposed LR, SVR, XGBoost and MLP models, it is clear that they are similar to each other in their predictions in reference to the figures regarding predicted value versus true value (Figures 5.1, 5.3, 5.5 and 5.7). By looking at the figures, it can be assumed that the bulk of the data is within the value span of 0.0 to 0.4 of the target label "*Sold load level*". As can be seen in the figures of all proposed models, there is a difficulty and almost impossibility in predicting the high outliers of higher than 0.6. As the outliers are rare in the data set, further fitting the models towards these can introduce and increase the risk of variance. Furthermore, the residual plots (Figures 5.2, 5.4, 5.6 and 5.8) are similar with no apparent trend.

In reference to the performance metrics, MSE, MAE and R^2 score, the proposed LR, SVR, XGBoost and MLP models predicts with results close to each other's. From the results of the statistical analysis the conclusion is drawn that there is no statistical significant difference amongst them, in regard to the prediction accuracy metrics. However, the SVR model underperforms compared to the other evaluated MLAs in regard to model fit and prediction time and the LR model is limited in regard to available hyperparameters. Apart from this, the remaining two models, XGBoost and MLP, have both great potential for further study, as both have many more hyperparameters which can be configured and further evaluated.

As seen in multiple previous studies [6, 12, 14], the prediction of a majority or all of the evaluated MLAs are comparable with no significant difference amongst themselves and with results close to each other's. The reasons behind these results could be many, the prediction achieved could be as good as possible for predicting the customer demand of in-flight meals based on the data that was available for the study. The hyperparameter grid search could have been insufficient for a subset of the evaluated MLAs. The method of preprocessing, feature engineering and model construction and selection could have been inadequate for the problem at hand and/or for a subset of the evaluated MLAs. In order to assess the reason for the obtained results with certainty, further research with various methods and larger time frames are necessary.

6.5 Ethical and Sustainability Aspects

When conducting research with real data, consideration on presentation and handling of the data set is a requirement. No sensitive data was used in this study as passenger information, current means of estimation and historical load level of flights was considered out of scope. The aim of this study is to evaluate multiple MLAs for predicting customer demand of in-flight meals in order to improve customer satisfaction and decrease the environmental impact of throwing away uneaten meals.

Increasing customer satisfaction means that all customers who wants to eat a meal should be able to. By increasing the amount of food on these flights, the passenger consummation could increase, leading to more waste and more fuel necessary for the flights which could potentially harm ecological sustainability. As the aim of the study and the goal of the MLAs are to meet that customer demand, both in increasing the amount of flights for certain flights as well as decreasing for other flights, this is not seen as a vital ethical issue.

By wish of the airline company, their name was withheld from the study and the pseudonym "airline company" has been used instead. As the study aims to evaluate multiple MLAs in comparison towards each other, the pseudonym was considered sufficient for both demonstrating the applicability of the presented MLA models and for further research. In agreement with Alan Bryman, when conducting quantitative research, anonymisation is a common tool for companies and individuals alike [79]. The pseudonym "airline company" is broad enough to minimise the risk of identifying the company while still being relevant for the research at hand.

Chapter 7

Conclusion

The aim of this study was to evaluate four regression MLAs for predicting customer demand of in-flight meals. In order to assess their applicability and performance, multiple measures and metrics were applied on a wide range of evaluated models. The conclusion follows that all four evaluated MLAs perform with results close to each other's in terms of prediction accuracy and with no significant difference against each other. The three more advanced MLAs (SVR, XGBoost and MLP) perform with similar results and marginally better in terms of prediction accuracy compared to the LR baseline. The SVR model underperformed however in terms of model fitting time and model prediction time and the LR model are left without available hyperparameters for further assessment and evaluation. This leaves the two models XGBoost and MLP, which have the best results in terms of prediction accuracy, acceptable results in terms of model fit and prediction time and both provides a wide range of possible hyperparameters and hyperparameter value spans for further evaluation. Furthermore, there is no apparent difference in terms performance metrics amongst the two MLAs.

All features in the available data set were used when creating the models and the feature significance analysis was performed before the model predictions and after the preprocessing of the data set. This means that the significance of the features is only compared towards each other's and the target labels and not towards the model and their predictions. The significance of each feature is evaluated in three metrics and visualised in a heat map. The most contributing and important feature for predicting customer demand of in-flight meals is the flight duration time, in terms of F-score, p-value, mutual information value and according to the feature correlation heat map.

The method presented in this study is designed for evaluating multiple MLAs and for data preprocessing. The different processes involved in the method could all influence the performance of the model in terms of both model prediction accuracy and fit/prediction time. The methods of normalisation and one-way-hot encoding of the data set performed were both considered necessary but could have been executed and presented in different constellations and manners. The study on related work might not have been sufficient for predicting customer demand of in-flight meals, resulting in selecting four MLAs based on false information. The data analysis of feature significance was only made post preprocessing and before model prediction. When assessing a features importance, the model at hand should be constructed in a forward selection manner with different constellation of said feature in comparison with others. This was considered as out of scope for this study due to a limited time frame.

The study presented a method of evaluating multiple regression MLAs for predicting customer demand of in-flight meals. The result show that XGBoost and MLP concludes in the best prediction accuracy and model fit and prediction time and are as such best suited for the problem. Furthermore, the most relevant attribute for the problem at hand, discovered in the analysis of the available data, was the scheduled flight duration time. This study serves as a foundation for further research for both evaluating multiple regression MLAs and for predicting customer demand.

7.1 Future Work

There are several approaches as to the means of carrying this research further. Two important limiting factors of this study was the time frame and the available data for model predictions. When conducting further research into the field, a narrower scope or a larger time frame are necessities for promising results. Furthermore, only a small subset of all available techniques of data preprocessing was considered for this study.

Future preprocessing could show if grouping/clustering of features has any significance on the model predictions. Either to group manually or by using methods such as K-means or K-prototypes to create clusters and smaller categories. Examples of such clusters could be of weekday clusters, scheduled time clusters and time of departure and arrival clusters. Either combined with this or done separately, by combining algorithmic feature selection with the knowledge of the data, feature weights could be introduced based on their importance. Further possible preprocessing would be to compare different methods of normalisation and for different methods try different value spans to further improve the model predictions. In this study only a $k = 5$ for the cross validation was considered, future studies could examine the effect of different k values as well as different sets of test/train ratio splits of the data set. Furthermore, the model hyperparameter selection phase was heavily dependent on the available values in the hyperparameter grid search. Future work could focus on evaluating a larger span of the value or alternatively, try different means of selecting and finding optimal hyperparameter configuration schemes.

Feature selection was completely considered as out of scope for this study but could prove to further improve performance results, in terms of both prediction accuracy and model fit and prediction time. Feature significance was however considered for this study and was conducted on the preprocessed data set before model prediction. To truly assess a features significance, the feature should be compared similar to the grid search on hyperparameters with all available subsets on the evaluated model.

Conclusively, future work should focus on further assessing the top performers of the MLAs considered for this study, namely SVR, XGBoost and MLP, in terms of feature selection and preprocessing techniques.

Bibliography

[1] Myungsook An and Yonghwi Noh. "Airline customer satisfaction and loyalty: impact of in-flight service quality". In: *Service Business* 3.3 (Sept. 2009), pp. 293–307. ISSN: 1862-8508. DOI: 10.1007/s11628-009-0068-4. URL: https://doi.org/10.1007/s11628-009-0068-4.

[2] Council of European Union. *Council regulation (EU) no 852/2004*. http://data.europa.eu/eli/reg/2004/852/oj. 2004.

[3] Airline company. *Introductory meeting*. Feb. 2019.

[4] Le Zhang, Jingbo Zhu, and Tianshun Yao. "An Evaluation of Statistical Spam Filtering Techniques". In: 3.4 (Dec. 2004), pp. 243–269. ISSN: 1530-0226. DOI: 10.1145/1039621.1039625. URL: http://doi.acm.org/10.1145/1039621.1039625.

[5] Yufeng Kou et al. "Survey of fraud detection techniques". In: *IEEE International Conference on Networking, Sensing and Control, 2004*. Vol. 2. Mar. 2004, 749–754 Vol.2. DOI: 10.1109/ICNSC.2004.1297040.

[6] Henrik Almér. "Machine learning and statistical analysis in fuel consumption prediction for heavy vehicles". MA thesis. KTH, School of Computer Science and Communication (CSC), 2015.

[7] K. C. Fuerstenberg, K. C. J. Dietmayer, and V. Willhoeft. "Pedestrian recognition in urban traffic using a vehicle based multilayer laserscanner". In: *Intelligent Vehicle Symposium, 2002. IEEE*. Vol. 1. June 2002, 31–35 vol.1. DOI: 10.1109/IVS.2002.1187923.

[8] Bryan Matthews et al. "Discovering Anomalous Aviation Safety Events Using Scalable Data Mining Algorithms". In: *Journal of Aerospace Information Systems* 10.10 (2013), pp. 467–475. DOI: 10.2514/1.I010080. eprint: https://doi.org/10.2514/1.I010080. URL: https://doi.org/10.2514/1.I010080.

[9] Ankur Kumar Jain, Manghat Nitish Menon, and Saurabh Chandra. "Sales Forecasting for Retail Chains". In: 2015.

[10] Tom M. (Tom Michael) Mitchell. *Machine learning*. eng. McGraw-Hill series in artificial intelligence. New York: McGraw-Hill, 1997. ISBN: 0-07-042807-7.

[11] Tianqi Chen and Carlos Guestrin. "XGBoost: A Scalable Tree Boosting System". In: *CoRR* abs/1603.02754 (2016). arXiv: 1603.02754. URL: http://arxiv.org/abs/1603.02754.

[12] Gregory F. Cooper et al. "An evaluation of machine-learning methods for predicting pneumonia mortality". In: *Artificial Intelligence in Medicine* 9.2 (1997), pp. 107–138. ISSN: 0933-3657. DOI: https://doi.org/10.1016/S0933-3657(96)00367-3. URL: http://www.sciencedirect.com/science/article/pii/S0933365796003673.

[13] V. Rodriguez-Galiano et al. "Machine learning predictive models for mineral prospectivity: An evaluation of neural networks, random forest, regression trees and support vector machines". In: *Ore Geology Reviews* 71 (2015), pp. 804–818. ISSN: 0169-1368. DOI: https://doi.org/10.1016/j.oregeorev.2015.01.001. URL: http://www.sciencedirect.com/science/article/pii/S0169136815000037.

[14] Raymond Salvador et al. "Evaluation of machine learning algorithms and structural features for optimal MRI-based diagnostic prediction in psychosis". In: *PLOS ONE* 12.4 (Apr. 2017), pp. 1–24. DOI: 10.1371/journal.pone.0175683. URL: https://doi.org/10.1371/journal.pone.0175683.

[15] J. Ling and J. Templeton. "Evaluation of machine learning algorithms for prediction of regions of high Reynolds averaged Navier Stokes uncertainty". In: *Physics of Fluids* 27.8 (2015), p. 085103. DOI: 10.1063/1.4927765. eprint: https://aip.scitation.org/doi/pdf/10.1063/1.4927765. URL: https://aip.scitation.org/doi/abs/10.1063/1.4927765.

[16] Nafisa Mahbub, Sanjoy Paul, and Abdullahil Azeem. "A neural approach to product demand forecasting". In: *International Journal of Industrial and Systems Engineering* 15 (July 2013), pp. 1–18. DOI: 10.1504/IJISE.2013.055508.

[17] Rosa Cantón Croda, Damian Gibaja, and Omar Caballero. "Sales Prediction through Neural Networks for a Small Dataset". In: *International Journal of Interactive Multimedia and Artificial Intelligence* InPress (Jan. 2018), p. 1. DOI: 10.9781/ijimai.2018.04.003.

[18] Tshilidzi Marwala Ishmael S. Msiza Fulufhelo V. Nelwamondo. "Water Demand Prediction using Artificial Neural Networks and Support Vector Regression". In: *Journal of Computers* 3 (Nov. 2008), pp. 1–8. URL: http://hdl.handle.net/10204/3374.

[19] Ana Lúcia Silva and Margarida GMS Cardoso. "Predicting supermarket sales: The use of regression trees". In: *Journal of Targeting, Measurement and Analysis for Marketing* 13.3 (Apr. 2005), pp. 239–249. ISSN: 1479-1862. DOI: 10.1057/palgrave.jt.5740150. URL: https://doi.org/10.1057/palgrave.jt.5740150.

[20] Ahmet Selman Bozkir and Ebru Akcapinar Sezer. "Predicting food demand in food courts by decision tree approaches". In: *Procedia Computer Science* 3 (2011), pp. 759–763.

[21] A.A. Levis and L.G. Papageorgiou. "Customer Demand Forecasting via Support Vector Regression Analysis". In: *Chemical Engineering Research and Design* 83.8 (2005), pp. 1009–1018. ISSN: 0263-8762. DOI: https://doi.org/10.1205/cherd.04246. URL: http://www.sciencedirect.com/science/article/pii/S0263876205727945.

[22] Chi-Jie Lu. "Sales forecasting of computer products based on variable selection scheme and support vector regression". In: *Neurocomputing* 128 (2014), pp. 491–499. ISSN: 0925-2312. DOI: https://doi.org/10.1016/j.neucom.2013.08.012. URL: http://www.sciencedirect.com/science/article/pii/S0925231213008850.

[23] Zong Woo Geem and William E. Roper. "Energy demand estimation of South Korea using artificial neural network". In: *Energy Policy* 37.10 (2009). Carbon in Motion: Fuel Economy, Vehicle Use, and Other Factors affecting CO2 Emissions From Transport, pp. 4049–4054. ISSN: 0301-4215. DOI: https://doi.org/10.1016/j.enpol.2009.04.049. URL: http://www.sciencedirect.com/science/article/pii/S0301421509003218.

[24] Markus Franke. "Competition between network carriers and low-cost carriers—retreat battle or breakthrough to a new level of efficiency?" In: *Journal of Air Transport Management* 10.1 (2004). Proceedings of the Hamburg Aviation Conference, 2003, pp. 15–21. ISSN: 0969-6997. DOI: https://doi.org/10.1016/j.jairtraman.2003.10.008. URL: http://www.sciencedirect.com/science/article/pii/S0969699703000826.

[25] M. I. Jordan and T. M. Mitchell. "Machine learning: Trends, perspectives, and prospects". In: *Science* 349.6245 (2015), pp. 255–260. ISSN: 0036-8075. DOI: 10.1126/science.aaa8415. eprint: http://science.sciencemag.org/content/349/6245/255.full.pdf. URL: http://science.sciencemag.org/content/349/6245/255.

[26] Trevor Hastie, Tibshirani Robert, and Friedman Jerome. *The Elements of Statistical Learning*. Springer, New York, NY, 2009. ISBN: 978-0-387-84858-7. URL: https://doi.org/10.1007/978-0-387-84858-7.

[27] D.C. Montgomery, E.A. Peck, and G.G. Vining. *Introduction to Linear Regression Analysis*. Wiley Series in Probability and Statistics. Wiley, 2012. ISBN: 9780470542811. URL: https://books.google.se/books?id=0yR4KUL4VDkC.

[28] Sergios Theodoridis. "Chapter 3 - Learning in Parametric Modeling: Basic Concepts and Directions". In: *Machine Learning*. Ed. by Sergios Theodoridis. Oxford: Academic Press, 2015, pp. 53–103. ISBN: 978-0-12-801522-3. DOI: https://doi.org/10.1016/B978-0-12-801522-3.00003-3. URL: http://www.sciencedirect.com/science/article/pii/B9780128015223000033.

[29] Alex J. Smola and Bernhard Schölkopf. "A tutorial on support vector regression". In: *Statistics and Computing* 14.3 (Aug. 2004), pp. 199–222. ISSN: 1573-1375. DOI: 10.1023/B:STCO.0000035301.49549.88. URL: https://doi.org/10.1023/B:STCO.0000035301.49549.88.

[30] Chih-Chung Chang and Chih-Jen Lin. "LIBSVM: A Library for Support Vector Machines". In: *ACM Trans. Intell. Syst. Technol.* 2.3 (May 2011), 27:1–27:27. ISSN: 2157-6904. DOI: 10.1145/1961189.1961199. URL: http://doi.acm.org/10.1145/1961189.1961199.

[31] Harris Drucker et al. "Support Vector Regression Machines". In: *NIPS*. 1996.

[32] Sergios Theodoridis. "Chapter 11 - Learning in Reproducing Kernel Hilbert Spaces". In: *Machine Learning*. Ed. by Sergios Theodoridis. Oxford: Academic Press, 2015, pp. 509–583. ISBN: 978-0-12-801522-3. DOI: https://doi.org/10.1016/B978-0-12-801522-3.00011-2. URL: http://www.sciencedirect.com/science/article/pii/B9780128015223000112.

[33] R. Bekkerman. *The present and the future of the kdd cup competition: an outsider's perspective*. Oct. 2015.

[34] Leo Breiman. *Classification and Regression Trees*. New York: Routledge, 2017. ISBN: 9781351460491. URL: https://doi.org/10.1201/9781315139470.

[35] Jerome H. Friedman. "Stochastic gradient boosting". In: *Computational Statistics & Data Analysis* 38.4 (2002). Nonlinear Methods and Data Mining, pp. 367–378. ISSN: 0167-9473. DOI: https://doi.org/10.1016/S0167-9473(01)00065-2. URL: http://www.sciencedirect.com/science/article/pii/S0167947301000652.

[36] W. Pitts W. McCulloch. "A logical calculus of ideas immanent in nervous activity". In: *Bulletin of Mathematical Biophysics* 5 (1943), pp. 115–133.

[37] Imad A Basheer and Maha Hajmeer. "Artificial neural networks: fundamentals, computing, design, and application". In: *Journal of microbiological methods* 43.1 (2000), pp. 3–31.

[38] Gareth James et al. *An introduction to statistical learning*. Vol. 112. Springer, 2013.

[39] Jundong Li et al. "Feature Selection: A Data Perspective". In: *ACM Comput. Surv.* 50.6 (Dec. 2017), 94:1–94:45. ISSN: 0360-0300. DOI: 10.1145/3136625. URL: http://doi.acm.org/10.1145/3136625.

[40] Isabelle Guyon and André Elisseeff. "An introduction to variable and feature selection". In: *Journal of machine learning research* 3.Mar (2003), pp. 1157–1182.

[41] Isabelle Guyon et al. *Feature extraction: foundations and applications*. Vol. 207. Springer, 2008.

[42] Michael P McLaughlin. *A compendium of common probability distributions*. Michael P. McLaughlin, 2001.

[43] Brian C Ross. "Mutual information between discrete and continuous data sets". In: *PloS one* 9.2 (2014), e87357.

[44] Alexander Kraskov, Harald Stögbauer, and Peter Grassberger. "Estimating mutual information". In: *Physical review E* 69.6 (2004), p. 066138.

[45] N Draper and H Smith. "Applied regression analysis: Wiley Interscience". In: *New York* (1981).

[46] Stanton A Glantz and Bryan K Slinker. *Primer of applied regression and analysis of variance*. Sirsi) i9780070234079. 1990.

[47] Airline company. *Data structure meeting*. Apr. 2019.

[48] Cort J Willmott and Kenji Matsuura. "Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance". In: *Climate research* 30.1 (2005), pp. 79–82.

[49] Christopher Manning, Prabhakar Raghavan, and Hinrich Schütze. "Introduction to information retrieval". In: *Natural Language Engineering* 16.1 (2010), pp. 100–103.

[50] David M. Allen. "Mean Square Error of Prediction as a Criterion for Selecting Variables". In: *Technometrics* 13.3 (1971), pp. 469–475. ISSN: 00401706. URL: `http://www.jstor.org/stable/1267161`.

[51] Richard R. Picard and R. Dennis Cook. "Cross-Validation of Regression Models". In: *Journal of the American Statistical Association* 79.387 (1984), pp. 575–583. ISSN: 01621459. URL: `http://www.jstor.org/stable/2288403`.

[52] Ron Kohavi et al. "A study of cross-validation and bootstrap for accuracy estimation and model selection". In: *Ijcai*. Vol. 14. 2. Montreal, Canada. 1995, pp. 1137–1145.

[53] Gregory W Corder and Dale I Foreman. *Nonparametric statistics for non-statisticians*. 2011.

[54] JL Hodges. "The significance probability of the Smirnov two-sample test". In: *Arkiv för Matematik* 3.5 (1958), pp. 469–486.

[55] S Cankurt and A Subasi. "Developing tourism demand forecasting models using machine learning techniques with trend, seasonal, and cyclic components". In: *Balkan Journal of Electrical and Computer Engineering* 3.1 (2015), pp. 42–49.

[56] Rand R Wilcox. "Some practical reasons for reconsidering the Kolmogorov-Smirnov test". In: *British Journal of Mathematical and Statistical Psychology* 50.1 (1997), pp. 9–20.

[57] Evie McCrum-Gardner. "Which is the correct statistical test to use?" In: *British Journal of Oral and Maxillofacial Surgery* 46.1 (2008), pp. 38–41.

[58] John Clapp and Carmelo Giaccotto. "Evaluating house price forecasts". In: *Journal of Real Estate Research* 24.1 (2002), pp. 1–26.

[59] Howard A Frank. "Municipal Revenue Forecasting with Time. Series Models: A Florida Case Study". In: *The American Review of Public Administration* 20.1 (1990), pp. 45–59.

[60] T Van Hecke. "Power study of anova versus Kruskal-Wallis test". In: *Journal of Statistics and Management Systems* 15.2-3 (2012), pp. 241–247.

[61] Eva Ostertagová, Oskar Ostertag, and Jozef Kováč. "Methodology and application of the kruskal-wallis test". In: *Applied Mechanics and Materials*. Vol. 611. Trans Tech Publ. 2014, pp. 115–120.

[62] Airline company. *Meeting number three*. May 2019.

[63] Weinan Zhang, Tianming Du, and Jun Wang. "Deep Learning over Multi-field Categorical Data". In: *Advances in Information Retrieval*. Ed. by Nicola Ferro et al. Cham: Springer International Publishing, 2016, pp. 45–57. ISBN: 978-3-319-30671-1.

[64] Yoshua Bengio et al. "Generalized denoising auto-encoders as generative models". In: *Advances in Neural Information Processing Systems*. 2013, pp. 899–907.

[65] Luai Al Shalabi, Zyad Shaaban, and Basel Kasasbeh. "Data mining: A preprocessing engine". In: *Journal of Computer Science* 2.9 (2006), pp. 735–739.

[66] Michael Waskom. *seaborn: statistical data visualization*. `https://seaborn.pydata.org/`. [Online; accessed 24-April-2019]. 2018.

[67] scikit-learn developers. *sklearn.feature_selection.mutual_info_regression*. https://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.mutual_info_regression. [Online; accessed 24-April-2019]. 2019.

[68] R Dennis Cook and Sanford Weisberg. *Residuals and influence in regression*. New York: Chapman and Hall, 1982.

[69] William S Cleveland. "Robust locally weighted regression and smoothing scatterplots". In: *Journal of the American statistical association* 74.368 (1979), pp. 829–836.

[70] Max Kuhn and Kjell Johnson. *Applied predictive modeling*. Vol. 26. Springer, 2013.

[71] Josef Perktold et al. *Linear Regression*. http://www.statsmodels.org/devel/regression. [Online; accessed 14-March-2019]. 2018.

[72] scikit-learn developers. *sklearn.svm.SVR*. https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor. [Online; accessed 15-March-2019]. 2019.

[73] Chih-Chung Chang and Chih-Jen Lin. *LIBSVM: A Library for Support Vector Machines*. Tech. rep. Department of Computer Science, 2013.

[74] Chi-Jie Lu and Yen-Wen Wang. "Combining independent component analysis and growing hierarchical self-organizing maps with support vector regression in product demand forecasting". In: *International Journal of Production Economics* 128.2 (2010). Supply Chain Forecasting Systems, pp. 603–613. ISSN: 0925-5273. DOI: https://doi.org/10.1016/j.ijpe.2010.07.004. URL: http://www.sciencedirect.com/science/article/pii/S092552731000229X.

[75] xgboost developers. *Introduction to Boosted Trees*. https://xgboost.readthedocs.io/en/latest/tutorials/model. [Online; accessed 8-April-2019]. 2019.

[76] Leo Breiman. "Random Forests". In: *Machine Learning* 45.1 (Oct. 2001), pp. 5–32. ISSN: 1573-0565. DOI: 10.1023/A:1010933404324. URL: https://doi.org/10.1023/A:1010933404324.

[77] keras-team. *The Sequential model API*. https://keras.io/models/sequential/. [Online; accessed 8-April-2019]. 2019.

[78] Google. *Keras*. https://www.tensorflow.org/guide/keras. [Online; accessed 8-April-2019]. 2019.

[79] Alan Bryman. *Social research methods*. Oxford university press, 2012.

Appendix A

Data Features

Table A.1 describes all 58 features included in the original raw data set presented from the airline company.

Table A.1: All features included in the raw dataset.

Feature	Description
Flight no	Flight number
Date	Date of departure, e.g. 24-DEC-16
yymm	Year and month of departure, e.g. 1612
STD	Scheduled time of departure
STA	Scheduled time of arrival
Weekday	Weekday of departure, 1-7
Departure station	IATA airport code of departure station
Arrival station	IATA airport code of arrival station
dep_arr	IATA airport code departure and arrival station pair
Route area	Route area in which the flight flies, 1E, 2D, 2E, 3D, 3E, 5I, 0U and 0A
Scheduled time	Scheduled flight time from departure to arrival
Class	Passenger class of the sales data, e.g. GO
Final pax	Final amount of passengers
No of sold/served Breakfast	Number of sold breakfast meals
No of sold/served Salad	Number of sold salad meals
No of sold/served Sandwich	Number of sold sandwich meals
No of sold/served Hot Snack	Number of sold hot snack meals
No of sold/served Dinner box	Number of sold dinner box meals
No of sold/served Day meal	Number of sold day meals
No of sold/served Fresh bite	Number of sold fresh bite meals
No of sold/served Polarrulle	Number of sold polarrulle meals
No of sold/served Lunch/Dinner	Number of sold lunch/dinner meals
BRML	Special food category
GFML	Special food category
CHML	Special food category
AVML	Special food category

continued . . .

. . . continued

Feature	Description
VGML	Special food category
DBML	Special food category
VLML	Special food category
NLML	Special food category
MOML	Special food category
LSML	Special food category
SFML	Special food category
HNML	Special food category
KSML	Special food category
LFML	Special food category
LCML	Special food category
BLML	Special food category
ADML	Special food category
KDML	Special food category
SAML	Special food category
TAML	Special food category
BBML	Special food category
DLML	Special food category
PZML	Special food category
VVML	Special food category
VTML	Special food category
EVML	Special food category
GCML	Special food category
GLML	Special food category
NPML	Special food category
LGML	Special food category
NOML	Special food category
SPML	Special food category
FIML	Special food category
MEML	Special food category
GHML	Special food category

Appendix B

Feature Significance

The results of the feature significance analysis on the preprocessed data set are presented in Table B.1.

Table B.1: Results of feature significance analysis on preprocessed data set.

Feature	F-score	p-value	mutual information value
Departure station_ABZ	48.03761994	4.22E-12	0.002263478
Departure station_LHR	24.81296384	6.34E-07	0.00185069
Departure station_BOO	28.37647599	1.00E-07	0.001893773
Departure station_BGO	134.7619464	4.03E-31	0.00315739
Departure station_ARN	10.22461168	0.001386528	0
Departure station_CPH	581.9957864	5.86E-128	0.004900085
Departure station_HAJ	48.64372319	3.10E-12	0
Departure station_AMS	25.53133306	4.37E-07	0.006354462
Departure station_GOT	205.1629837	1.87E-46	0.006099888
Departure station_HAM	151.0257577	1.14E-34	0.00578085
Departure station_OSL	374.6797704	3.27E-83	0.010459075
Departure station_VBY	128.7819059	8.13E-30	0.006592781
Departure station_KKN	7.859180933	0.005057997	0
Departure station_NCL	4.000668151	0.045486853	0.00508015
Departure station_BRU	0.300537529	0.58354773	0
Departure station_OSD	13.03055698	0.000306706	0
Departure station_TLL	126.0247243	3.25E-29	0.004448873
Departure station_KEF	64.98665399	7.68E-16	0
Departure station_SFT	0.814509393	0.36679412	0.002599416
Departure station_LLA	0.160451925	0.688742246	0.001610827
Departure station_AGP	1725.386988	0	0.005171288
Departure station_KRS	44.71980271	2.29E-11	0.002590431
Departure station_MXP	57.28718432	3.82E-14	0.003844733
Departure station_HEL	186.047686	2.69E-42	0.003551294
Departure station_FRA	1.943173624	0.163329665	0.005610534
Departure station_TRD	130.9381293	2.75E-30	0
Departure station_SVG	42.82908478	6.02E-11	0.003296474

continued ...

. . . continued

Feature	F-score	p-value	mutual information value
Departure station_PMI	1334.030378	8.37E-289	0.004689379
Departure station_GVA	105.3309441	1.08E-24	0
Departure station_PLQ	36.64403789	1.43E-09	0
Departure station_LED	6.268220242	0.012294962	0.003273211
Departure station_AGH	0.762380881	0.382588048	0
Departure station_ALC	1214.558358	2.14E-263	0.009416424
Departure station_UME	32.3225971	1.31E-08	0.005449336
Departure station_MAN	8.412254553	0.003728392	0
Departure station_TOS	13.85030534	0.00019815	0.003879636
Departure station_SPU	773.2584084	4.48E-169	0.007684646
Departure station_POZ	16.77974953	4.20E-05	0.00347101
Departure station_TKU	88.61297023	4.97E-21	0.006301634
Departure station_SKG	24.52096903	7.37E-07	0.001241215
Departure station_MLA	370.5654088	2.54E-82	0.001311041
Departure station_STR	25.01341116	5.71E-07	0
Departure station_FCO	445.9465174	1.29E-98	0.005144497
Departure station_KRN	9.378229945	0.002196773	0.002267709
Departure station_KLR	32.79499195	1.03E-08	0
Departure station_MUC	7.206738702	0.007265085	0.001743127
Departure station_TXL	18.75010523	1.49E-05	0.000798339
Departure station_RNB	5.397839963	0.020165106	0
Departure station_RIX	58.45891781	2.11E-14	0.004582105
Departure station_DUS	37.99619808	7.13E-10	0.0049894
Departure station_ZRH	18.97906395	1.32E-05	0.001274129
Departure station_BLQ	2.65502725	0.103228382	0.00097074
Departure station_HAU	23.31783363	1.38E-06	0.004818401
Departure station_ATH	213.1537335	3.43E-48	0.002353612
Departure station_AAL	8.635516163	0.003298008	0
Departure station_MMX	16.12296736	5.94E-05	0.000464345
Departure station_NCE	322.0808727	7.96E-72	0.005470265
Departure station_CDG	134.5644458	4.45E-31	0.005289434
Departure station_DBV	151.7743774	7.84E-35	0
Departure station_DUB	204.7248553	2.33E-46	0.005324994
Departure station_AES	67.87500283	1.78E-16	0.002366735
Departure station_PUY	6.423040066	0.011267506	0
Departure station_GDN	87.42010282	9.07E-21	0
Departure station_SDL	1.343803653	0.246369039	0.001664614
Departure station_EDI	13.52207053	0.000235985	0
Departure station_VAA	78.38691111	8.70E-19	0.006167584
Departure station_LIN	126.5190237	2.54E-29	0.007208067
Departure station_WAW	39.55989486	3.20E-10	0.004839973
Departure station_AAR	6.092825138	0.013576032	0.002975316
Departure station_VNO	76.85542818	1.89E-18	0.001543629
Departure station_BHX	11.82872178	0.000583647	0.00028397

continued . . .

...continued

Feature	F-score	p-value	mutual information value
Departure station_LYR	139.096811	4.56E-32	0
Departure station_EVE	1.499867219	0.220696734	0.000647674
Departure station_BRE	0.993766025	0.318827795	0.004494523
Departure station_OER	3.285031736	0.06991941	0.002374692
Departure station_TMP	74.03506471	7.86E-18	0.000641414
Departure station_BLL	39.79026105	2.85E-10	0.002330674
Departure station_BCN	499.5676529	3.43E-110	0.00614378
Departure station_PRG	0.286427309	0.592521182	6.96E-05
Departure station_FAO	616.7488036	1.93E-135	0
Departure station_WRO	19.37213053	1.08E-05	0.000113518
Departure station_MOL	16.24955907	5.56E-05	0.003595457
Departure station_OTP	1.037674021	0.308367271	0.000456952
Departure station_GZP	399.0181808	1.78E-88	0.004989631
Departure station_JMK	111.906911	3.95E-26	0
Departure station_KSU	17.98030914	2.24E-05	0
Departure station_TLN	0.042526007	0.836621075	0
Departure station_LPI	5.842100392	0.015650181	0.004047374
Departure station_KRK	1.503801045	0.220092379	0.001916274
Departure station_LPA	1189.989691	3.66E-258	0.003336668
Departure station_VCE	17.50774332	2.87E-05	0
Departure station_NAP	60.09716339	9.17E-15	0
Departure station_ALF	0.976128414	0.323160605	0
Departure station_CHQ	34.41814951	4.47E-09	0.004605407
Departure station_SZG	6.846297181	0.008884805	0.005197001
Departure station_GOA	0.306140205	0.580060543	0
Departure station_BIQ	21.59670398	3.37E-06	0
Departure station_LIS	244.0527913	6.64E-55	0.001323849
Departure station_KUN	0.649480571	0.420301732	0
Departure station_OLB	55.9054667	7.71E-14	0
Departure station_VIE	0.12763314	0.720900106	0
Departure station_FUE	24.56830059	7.19E-07	0
Departure station_BUD	3.325383672	0.068223205	0
Departure station_MPL	2.124329986	0.144981543	0
Departure station_TFS	22.98786189	1.63E-06	0
Departure station_OUL	0.348408698	0.555017597	0
Departure station_BEY	13.86536937	0.000196568	0
Departure station_CAG	18.8988743	1.38E-05	0
Departure station_INN	23.92318595	1.01E-06	0.002867325
Departure station_PSA	48.90243751	2.72E-12	0
Departure station_PRN	2.262229876	0.132568469	0
Departure station_RHO	2.75849251	0.096744725	0.00295185
Departure station_PMO	35.84775373	2.15E-09	0.004511078
Departure station_JTR	69.78589553	6.75E-17	0.002250431
Departure station_SVO	0.22442339	0.635691762	0

continued ...

…continued

Feature	F-score	p-value	mutual information value
Departure station_FNC	83.40802457	6.88E-20	0
Departure station_VDA	0.345368867	0.556748755	0
Departure station_CMF	0.854017896	0.355421993	0
Departure station_SNN	3.175832439	0.074740315	0.001394292
Departure station_TLV	0.043949693	0.833947969	0.000565959
Arrival station_OSL	36.06099788	1.92E-09	0.004360536
Arrival station_ARN	13.17098736	0.000284559	0.001910005
Arrival station_GOT	110.9138937	6.52E-26	0.004564715
Arrival station_WAW	37.16886294	1.09E-09	0.002976923
Arrival station_CPH	572.3814688	6.90E-126	0.008509643
Arrival station_AMS	17.98605872	2.23E-05	0.001913244
Arrival station_SVG	134.3627784	4.92E-31	0.000553665
Arrival station_AGH	2.219320475	0.136298437	0.002230187
Arrival station_MUC	0.353128063	0.552350031	0
Arrival station_PMI	587.3583264	4.10E-129	0.004173921
Arrival station_TLL	71.86279843	2.36E-17	0
Arrival station_MAN	46.69677495	8.37E-12	0.006186626
Arrival station_KRK	12.375269	0.000435388	0.000262563
Arrival station_AGP	1361.579072	1.18E-294	0.006138111
Arrival station_LHR	556.7753166	1.59E-122	0.012373638
Arrival station_BOO	0.966648648	0.325521495	0.005402404
Arrival station_BGO	96.98220238	7.29E-23	0.001997343
Arrival station_GDN	61.38462299	4.77E-15	0.002461321
Arrival station_RNB	22.84808445	1.76E-06	0
Arrival station_LYR	3.604051452	0.057643867	0.001648884
Arrival station_ABZ	26.8925607	2.16E-07	0
Arrival station_KLR	34.36558597	4.59E-09	0.004064399
Arrival station_LLA	15.89309725	6.71E-05	0
Arrival station_EVE	2.400819998	0.121277087	0.002972959
Arrival station_BLL	37.68731125	8.36E-10	0.003587043
Arrival station_HAJ	50.78197305	1.04E-12	0
Arrival station_TOS	196.2229173	1.65E-44	0.003929148
Arrival station_PLQ	32.05926014	1.50E-08	0.003667914
Arrival station_POZ	9.656335318	0.001887902	0.000531491
Arrival station_SPU	164.9744504	1.04E-37	0
Arrival station_ZRH	35.57272945	2.47E-09	0.00136069
Arrival station_TRD	95.67387308	1.41E-22	0.00471875
Arrival station_PRG	34.91455271	3.46E-09	0.001159491
Arrival station_VAA	53.48313863	2.64E-13	0
Arrival station_HAM	121.8344615	2.68E-28	0
Arrival station_MMX	4.062764367	0.043843251	0.001086625
Arrival station_LED	17.1651753	3.43E-05	0.002040266
Arrival station_VNO	42.39504916	7.52E-11	0.004480445
Arrival station_STR	20.54249669	5.84E-06	0

continued …

...continued

Feature	F-score	p-value	mutual information value
Arrival station_DUS	19.49469677	1.01E-05	0
Arrival station_MXP	46.86412728	7.68E-12	0.003097199
Arrival station_AES	25.26150445	5.02E-07	0
Arrival station_VBY	119.6105781	8.19E-28	0.00059572
Arrival station_UME	26.15446006	3.16E-07	0
Arrival station_OSD	5.611902634	0.017842118	0.000613244
Arrival station_AAL	8.033632212	0.004593242	0
Arrival station_NCE	443.0832163	5.35E-98	0.004215107
Arrival station_NCL	0.224647139	0.63552339	0
Arrival station_GVA	91.39280489	1.22E-21	0.001593467
Arrival station_SFT	0.978457816	0.322583942	0
Arrival station_KEF	176.5311322	3.17E-40	0.003529391
Arrival station_OER	8.806996476	0.003001993	0.002143965
Arrival station_AAR	14.21097092	0.000163576	0.005237186
Arrival station_BRU	1.798929808	0.179847146	0
Arrival station_KRN	11.2869453	0.000781039	0.007708202
Arrival station_TXL	1.751376054	0.185709018	0.000322391
Arrival station_DUB	300.7958268	3.26E-67	0.000111484
Arrival station_TMP	70.78251895	4.08E-17	0.001834748
Arrival station_LIN	279.0149837	1.72E-62	0
Arrival station_TKU	90.51001576	1.91E-21	0.003949531
Arrival station_RIX	30.95987217	2.65E-08	0.0008954
Arrival station_SDL	15.27164153	9.32E-05	0
Arrival station_EDI	74.49814793	6.22E-18	0
Arrival station_CDG	143.5146206	4.96E-33	0.00063557
Arrival station_BRE	6.171688094	0.01298385	0.001909741
Arrival station_HEL	88.74252872	4.65E-21	0
Arrival station_ALF	23.92845324	1.00E-06	0.000168703
Arrival station_KRS	0.527951416	0.467472315	0.004487379
Arrival station_FCO	250.8346327	2.24E-56	0
Arrival station_BLQ	3.014331711	0.082536741	0.003705498
Arrival station_FRA	0.756591834	0.384400784	0.0030912
Arrival station_KKN	18.59613498	1.62E-05	0
Arrival station_LPA	2521.52762	0	0.008276598
Arrival station_BCN	276.4150772	6.30E-62	8.17E-05
Arrival station_BHX	3.611973332	0.057369934	0
Arrival station_ALC	625.9854481	1.98E-137	0.005658236
Arrival station_SKG	27.817783	1.34E-07	0
Arrival station_VCE	39.30655987	3.65E-10	0
Arrival station_FAO	273.1073978	3.29E-61	0.000374019
Arrival station_KUN	0.767386386	0.381030415	0
Arrival station_OTP	4.836702774	0.027864021	0
Arrival station_WRO	8.199714576	0.00419119	0.003127588
Arrival station_KSU	13.89153753	0.000193851	0

continued ...

. . . continued

Feature	F-score	p-value	mutual information value
Arrival station_ATH	61.82689622	3.81E-15	0
Arrival station_LPI	7.54380406	0.006023522	0
Arrival station_GZP	397.5058989	3.78E-88	0.001907315
Arrival station_BIQ	20.23179134	6.87E-06	3.40E-05
Arrival station_DBV	17.20003927	3.37E-05	0
Arrival station_GOA	3.315888004	0.068618358	0
Arrival station_MOL	10.36799365	0.001282889	0.002391489
Arrival station_OLB	9.064684155	0.002607049	0.001264601
Arrival station_HAU	14.02655949	0.00018042	0
Arrival station_JTR	0.345368867	0.556748755	0
Arrival station_CHQ	122.3473842	2.07E-28	0
Arrival station_VIE	0.565285167	0.452141602	0
Arrival station_MLA	100.4930034	1.24E-23	0
Arrival station_FUE	90.77183841	1.67E-21	0
Arrival station_SVO	7.151478976	0.007492304	0
Arrival station_TLN	0.579548003	0.446492618	0
Arrival station_SZG	0.025270909	0.873694427	0
Arrival station_PRN	1.111715118	0.291713786	0
Arrival station_PMO	14.34853747	0.00015205	0.000476901
Arrival station_LIS	52.84280627	3.66E-13	0.000239494
Arrival station_PSA	18.49180109	1.71E-05	0.001583028
Arrival station_MPL	4.235939668	0.039581138	0.005108295
Arrival station_OUL	0.395864312	0.529235326	0.001137815
Arrival station_VDA	1.036154349	0.308721776	0
Arrival station_BUD	0.296354185	0.586179178	0.004525622
Arrival station_SNN	0.345368867	0.556748755	0
Arrival station_FNC	95.25944182	1.74E-22	0
Arrival station_PUY	20.61863396	5.62E-06	0
Arrival station_INN	3.519589219	0.060652461	0.000695094
Arrival station_VRN	0.701808982	0.402179946	0
Arrival station_SJJ	2.322808847	0.127495045	0
Arrival station_BEY	0.022515584	0.880724156	0.000545355
Arrival station_RHO	7.078244216	0.007804675	0.002574364
Route area_2E	3129.218269	0	0.022236552
Route area_3E	1997.537342	0	0.02087806
Route area_2D	130.9633281	2.72E-30	0.004944416
Route area_5I	1535.452449	0	0.02187832
Route area_3D	376.7096679	1.19E-83	0.012608503
Route area_1E	289.683948	8.36E-65	0.005262761
STD	1300.898789	9.19E-282	0.065970542
STA	14.94070247	0.000111063	0.052949308
Weekday	150.4264244	1.54E-34	0.013723969
Scheduled time	36277.26887	0	0.240436799
Day	1.664883122	0.196950412	0.000363552

continued . . .

. . . continued

Feature	F-score	p-value	mutual information value
Month	4.316833758	0.037741436	0.007917453
Year	71.89912095	2.32E-17	0.001903481

TRITA -EECS-EX-2019:487