Norwegian School of Economics
Bergen, Spring 2022

The impact of implementing a fleet allocation and scheduling decision support system

Development and application of a machine learning event impact analysis framework

Stanisław Zawieja

Supervisor: Julio C. Góez

Master thesis, Economics and Business Administration

NORWEGIAN SCHOOL OF ECONOMICS

Preface

First and foremost, I would like to extend my gratitude to Julio Góez for inspiring discussions, precise comments, and sharing my excitement for the subject. I would like to thank Erik Fritz Loy, Bernhard Hafting, Julie Schasler, and all others at Dataloy Systems for providing me with the topic for this thesis, invaluable maritime shipping insights, and a warm and supportive atmosphere throughout the whole process. I would also like to thank Jacek Wallusch for generously sharing his data science knowledge. In addition, I wish to thank the shipping company which decided to remain anonymous for allowing me to utilize their private maritime data.

I chose my thesis to be set in the maritime shipping industry, which is fundamental both for global trade and for the local economy here in Bergen, Norway. However, I aimed to write a data science paper which is set within shipping, not a shipping paper which employs data science. Additionally, much like my major of Business Analytics itself, I wanted my thesis to also transcend a single industry or use. I aimed for researching a solution which can be narrow enough to solve a specific business problem, but broad enough to be easily reusable and applicable in completely different contexts.

After my partner company presented me with their research problem, I understood that this is the perfect opportunity to write such a paper. I realized that measuring the effect of maritime software is still in uncharted waters of data science. Therefore, I applied and expanded on existing event impact analysis research in an attempt to fill those blank spaces.

Norwegian School of Economics

Bergen, June 2022

Stanisław Zawieja

Abstract

Maritime scheduling software is a fairly new and dynamically growing industry. While statistical methods for event impact analysis are well research, they have yet to be applied to this new area. This master thesis investigates the impact of implementing a fleet allocation and scheduling software by a maritime shipping company. Ideas from existing research by Wang et al. (2019) are applied in order to assess event impact on Vessel Weight Utilization, defined as the ratio of cargo weight and the deadweight of the ship, which is a proxy for total cargo carrying capacity. Random Forest is used to predict the "would be" performance of the KPI in a counterfactual scenario in which the software is never introduced. The difference between the actual KPI time series and the "would be" scenario quantifies the software impact.

Furthermore, this paper expands on the existing framework by proposing a way to use Random Forest to make two predictions of the KPI, one for the factual scenario and one for the counterfactual scenario. This allows not only for calculating software impact, but also for prediction distributions to be compared using, among others, kernel density plots and the Kolmogorov–Smirnov test. OLS models are used as a naive benchmark to check the validity of the methods used.

Results suggest that implementing the fleet allocation and scheduling software had a slight effect on the shape of the distribution, but ultimately did not have a visible effect on mean Vessel Weight Utilization over a 2 year period after software implementation. This can be viewed as a positive outcome given that the focus of the Decision Support System during the studied period was on increasing user experience, rather than fleet plan mathematical optimization. The research results indicate that switching to a digital tool marketed as more scalable and increasingly optimization-based does not create a substantial operational risk for the maritime shipping company as measured by the tracked KPI.

Keywords – Event Impact Analysis, Key Performance Indicator, Machine Learning, Random Forest, Decision Support System, Maritime software effect, Fleet scheduling, Maritime shipping

Contents

List of Figures

List of Tables

Abbreviations used

Abbreviation	**Full form**
ARIMA	Autoregressive Integrated Moving Average
DSS	Decision Support System
FAS	Fleet Allocation and Scheduling
IQR	Interquartile Range
KPI	Key Performance Indicator
KS test	Kolmogorov - Smirnov test
MAE	Mean Absolute Error
ML	Machine Learning
MSE	Mean Square Error
NA	Not Available (missing value)
NTNU	Norwegian University of Science and Technology
OLS	Ordinary Least Squares
PDF	Probability Density Function
RMSE	Root Mean Square Error
RF	Random Forest
TCE	Time Charter Equivalent
UX	User Experience
VWU	Vessel Weight Utilization

1 Introduction

1.1 Assessing the risk of implementing new solutions

This study aims to perform an event impact analysis to measure the effect of a fleet allocation and scheduling software on a maritime transportation company's relevant KPI. The study outcome can be viewed in the context of an assessment of business operational risk for the maritime shipping company which decides to switch to using the new software.

Historically, profound understanding of this kind of operational risk used to be neglected. However, in the last two decades this topic rose to prominence (Moosa, 2007). Nowadays, operational risk is a big concern for businesses. (Bonet et al., 2021)

Bonet et al. (2021) discuss that while usually companies assess their operational risk based on their internal data, there are other factors that can affect this assessment. Those factors include scenario analyses, which can provide insights about the operational risk in a situation which did not necessarily take place but could possibly happen. This master thesis makes use of a similar idea of creating "would be" scenarios. By comparing them with each other and with the real-world historical data, it provides insights about risk resulting from switching to a digital DSS. A deeper understanding of this kind of operational risk is relevant for several players. In the broad context, it is relevant for the maritime shipping industry and the maritime software industry. In the specific context it is relevant for the partner company and the customer shipping company who provided information and relevant data for conducting this analysis.

1.2 Moving towards a data-driven industry

As maritime shipping becomes increasingly more digitalized and moves towards becoming a data driven industry, maritime KPI tracking and analysis play a key role in this industry-wide transformation. Already in 2013, Duru et al. mentioned that "in the last few years, the use of Key Performance Indicators (KPIs) is a popular and trending practice in the business" (Duru et al., 2013). Today this practice of tracking relevant maritime data continues to increase as maritime software companies, such as Veson Nautical, Q88, or Dataloy Systems AS, offer maritime software solutions and market their solutions as data driven (Veson Nautical, 2022a; Digital Ship, 2021).

It is not uncommon for companies in the maritime software industry to claim to provide data-driven maritime scheduling DSS solutions which achieve positive effects (Veson Nautical, 2022b). However, to the knowledge of the author of this study, such claims appear to be unsupported, without clear ties to scientific methods or backing research. In some cases, surface level qualitative assessment is presented as quantitative research (Veson Nautical, 2022b). It is easy to claim and promote positive effects of maritime scheduling DSS but it is significantly more difficult and complex to actually achieve numerical results in assessing such software (Diz et al., 2014; Fagerholt, 2004; Fagerholt and Lindstad, 2007). While there exists research focusing on single-use solutions, it is difficult to find any descriptions of accomplishing the task of measuring the value creation of a flexible maritime software designed to be used by many maritime transportation companies for an extended period.

1.3 Relevance of this thesis

This paper is written in cooperation with Dataloy Systems AS (later referred to as Dataloy), which presented the author with a specific business case. Dataloy would like to understand how the implementation of FAS, their product, really affects the business risk for their customer. While subjective perception of business risk could of course be measured by qualitative interviews, it is significantly more difficult to assess the underlying risk itself numerically. As mentioned by Wang et al. (2019), "determining whether an event has positive, negative or no effect on KPI is not a trivial task". Nonetheless, the goal of this analysis is to quantify such event effect to provide the partner company with a more tangible and objective statistic revealing whether the implementation of their product had a positive, negative, or no effect on a chosen indicator. This analysis can be useful as internal feedback concerning one of the company's key products, which in turn has the potential to influence Dataloy's marketing and pricing strategy.

The author's previous professional experience within maritime data analytics suggests that spot maritime shipping is heavily driven by market forces and a subject to highly variable TCE rates. Furthermore, the spot market rates generally lack regular seasonal patterns. For this reason, it is considered difficult for maritime software providers to convince shipping companies that their product adds value, or that the benefits of implementing their solution outweigh the business risks. This problem was already visible when Marintek, an NTNU research institute, tried to market their maritime DSS in the early 21st century (Fagerholt, 2004). According to the experience of the author of this thesis gained by working with a maritime software company and interviewing maritime software professionals, this problem is still relevant to this day. This thesis aims to address that issue. In Section 1.2, I elaborated on the trend of maritime software companies using "data" as a catch phrase while describing their success stories. However, it is not obvious that those claims are based on any numerical analysis results. This research is relevant for the maritime software industry, which can attempt to derive business value from maritime data analytics research which does provide the needed numerical results.

Moreover, this thesis can be of interest for the anonymous partner shipping company, because it provides an analysis of how a product their purchased, and by extension an operational framework they implemented, affected their business. Similarly, the same logic of relevance of this research can be extended to maritime shipping companies in general, especially the ones which are considering purchasing FAS or similar software.

Additionally, in the maritime software industry, there exists a prevailing idea that the maritime shipping industry can be characterized as conservative and historically reluctant to adapt new ways of doing business. Fleet scheduling, which is an essential part of operations of maritime shipping companies, has traditionally been done using a system of spreadsheets rather than with the help of integrated maritime software solutions. As fleet allocation and scheduling decision support systems are being developed and marketed, there exists an industry-wide need to assess the efficiency of those solutions as opposed to traditional, manual methods. This paper aims to contribute to closing this knowledge gap. The framework and methods used in this paper can be generalized to provide insight for both professional and academic maritime software research.

To conclude, by shedding light on the effects of maritime software, this paper not only contributes to the general body of knowledge encompassing the maritime transportation and maritime software industries, but also directly addresses a specific business need.

1.4 Dataloy's FAS

This research project is essentially an event impact analysis where the goal is to quantify the impact of the event, implementing maritime software, on the business of the shipping company implementing it. This section provides details about the maritime software studied.

FAS is a software which assists schedulers in creating fleet plans. A planner can use FAS to create a new sea voyage, allocate cargoes on that voyage, edit specific information concerning each cargo and voyage, exchange cargoes between voyages, and shift voyages from one vessel to another to finalize a schedule for a whole fleet. Several fleet schedule scenarios can be created and compared before a scheduler can "nominate" a voyage when they are sure that a given voyage transporting given cargo will be done by a specific ship.

This basic example of a scheduler's workflow is facilitated by FAS automatically calculating and adjusting information such the quantity of cargo on board, the dates of events like a voyage start or a port call, or an indication of a vessel missing the laycan window. It enables processes which were traditionally done using spreadsheets to be performed inside a web application where users have access to an overview including a list unallocated cargoes and other information related to maritime planning. Users are also able to test out different fleet schedule scenarios, and can collaborate with others.

According to Dataloy's website, their solution "encourages the planner's soft skills and expertise" (Dataloy Systems AS, 2022). The company's philosophy of prioritizing UX and creating a tool which supports the work of human fleet planners, before gradually implementing more advanced solutions like fleet optimization algorithms is in line with the findings of NTNU's Marintek institute (Fagerholt, 2004; Fagerholt and Lindstad, 2007). This connection is explained in more detail in Section 3.1 On the other hand, because of this philosophy FAS did not include robust mathematical optimization algorithms for optimizing fleet schedules in the studied period.

Additionally, an important aspect of the decision support system is providing the user with an application which, according to Dataloy's claims, is a simpler, more scalable, time-saving solution as compared to traditional manual methods. (Dataloy Systems AS, 2022). If that is the case, it is important for the shipping company to know whether that solution will not decrease the quality of the work of schedulers. Therefore, the time and money investment related to switching to a digital DSS solution can be perceived as an operational risk by maritime transportation companies. Assessing the impact of software implementation on a relevant shipping KPI quantifies that risk.

2 Research question formulation

Two-part research question

The goal of this research is to quantify the impact of FAS on a maritime shipping company, which corresponds to operational business risk for that company. The research question formulated to solve that business problem can be divided into two parts.

1. Did implementing the software have any negative or positive effect on the value of the tracked shipping KPI?
   (a) Did software implementation have a statistically significant effect on the tracked KPI?
   (b) How did the tracked KPI change on average after software implementation, if it changed at all?
2. Did software implementation have an effect on the distribution of the KPI?

3 Literature review

3.1 Previous maritime software effectiveness research

The literature on decision support systems for fleet scheduling is limited. There is a wealth of research concerning maritime shipping optimization on a case-by-case basis. That is, one-time scientific projects aiming to solve a very specific problem or optimize operations for a single maritime shipping company. (Diz et al., 2014) However, it is difficult to find many descriptions of long term, flexible DSS solutions which could be used to support fleet allocation and scheduling of many companies over a long period.

One exception is Turborouter, a DSS developed in the early 21st century by Marintek, a research institute at NTNU. The researchers published a few papers about it, but stopped publishing after 2008 (Marintek, 2008). Turborouter included dynamic port to port calculation, a schedule calculator with multiple ports and time in each port, a cargo allocation tool, real time satellite positions of the vessels, a fuel consumption calculator, and optimization algorithms for vessel fleet scheduling. The scientists tried to approach the problem purely from the optimization perspective, but did not succeed long term from the business perspective. Through conversations with shipping experts within Dataloy I found that shipping companies can be reluctant to digitalize due to a conservative mindset. Other barriers can include lack of funding, no proven RoI, and cybersecurity concerns.

The two decades old findings of the NTNU scientists are in line with Dataloy's underlying philosophy. Marintek found that while algorithmic solutions exist, it was difficult to convince shipping companies to replace spreadsheets with modern solutions. The key finding of Marintek was that the focus should be shifted from an optimization tool to a more user friendly decision support system. They recommended "developing the DSS in small steps" with feedback loops at each step, as well as "selling vessel position reports as a product based on internet subscription". (Fagerholt, 2004) Marintek stressed the importance of putting the customer in focus by allowing a manual cargo assignment function while the algorithmic solution was to remain a suggestion.

To the disappointment of the author of this thesis, it appears that the size of the body of knowledge in this topic did not undergo any drastic changes over the years.

According to Diz et al. (2014) “Although improved and detailed mathematical models and optimization algorithms to support ship scheduling decision making are available, few studies demonstrate the successful applications of such tools in real-life cases.”

The shortage of successful applications of fleet allocation and scheduling decision support systems makes it even more difficult to measure their value creation. Diz et al. (2014) claim that a DSS designed for PETROBRAS resulted in 7.5% reduction in operational costs, but this was also case-based optimization. According to NTNU’s Marintek, “in general, DSS benefits are often difficult to measure” (Fagerholt, 2004). The institute remained convinced of this up until 2007, when the researchers reiterated that "in general improvements and benefits resulting from the use of DSSs are often difficult to measure” (Fagerholt and Lindstad, 2007). However, Fagerholt and Lindstad describe that their optimization software gave their customers cost reductions of 4%-5% as compared with manual planning. Those considerable savings were driven by increased fleet utilization, and with the shipping company being able to transport a few additional spot cargoes over a relatively short period (Fagerholt and Lindstad, 2007). The researchers also reported that the time spent on schedule planning experienced a significant reduction, which allowed for planners to engage in other creative tasks (Fagerholt and Lindstad, 2007). The NTNU institute stopped publishing about Turborouter a year later, and I am not aware of any other previous research which describes accomplishing the task of measuring the value creation of a flexible maritime software designed to be used by many companies.

3.2 Previous event impact analysis research

Previous event impact analysis research can be divided into five categories. Each of the approaches is subject to different limitations, and contributes to the general body of knowledge on this topic under a different set of assumptions.

First, impact of an event can be analyzed by a comparison of KPI values before and after the event. This method was applied by Mbugua et al. (1995) to research the impact of removal of registration fees in public healthcare on the use of health services by vulnerable groups of people in Kenya. The researchers gather longitudinal data from 9 months before until 2 months after the policy change and apply statistical tests to compare the two samples. Mbugua et al. (1995) achieve results showing a difference in the number of people who use health care services between the two time periods, however this method does not assure that the measured change indeed happened due to the studied event. While comparing before and after data an assumption has to be made that the change between the two data samples was not a result of time series characteristics such as seasonality or trend. If seasonality or trend are present in the data, the choice of the observation window could impact the analysis results.

Second, ARIMA time series modeling can be used to evaluate how a KPI has changed over time. This traditional econometric method was applied by Koski et al. (2007) to study impact of policy changes on alcohol consumption in Finland. An even earlier study proposed modeling a time series using a differential approach to analyze smog data in Los Angeles (Box and Tiao, 1975). While a difference equation is not synonymous with an ARIMA model, ARIMA approach includes a differencing term. Here, the main idea can be to fit the model on pre-event time series data, and use it to make a prediction over a scenario data set which includes the studied event. Using this method for event impact analysis was criticized by Lagarde (2012) who points out that it can be problematic to select the best-fitting ARIMA model to appropriately model the time series. Furthermore, Lagarde (2012) mentions that ARIMA modeling might not be appropriate in cases in which there are not enough data points, since this technique works best for long time series. Additionally, it is worth to indicate that ARIMA makes the most sense for data where seasonality and trend can be expected, such as macroeconomic indicators studied by Koski et al. (2007) and Box and Tiao (1975). A measured divergence from the expected seasonality and trend could be evidence for a public policy causing change. However, many business KPIs, such as shipping KPIs, are not characterized by clear seasonal and long term trend patterns. Therefore ARIMA modeling is not appropriate for more variable KPIs, since it is unfeasible to measure divergence from a seasonal pattern without being able to convincingly model the seasonal pattern in the first place.

Third, segmented linear regression is presented by Lagarde (2012) as a method alternative to ARIMA modeling. The author introduces a binary variable taking on a value of 1 for observations for which a policy change took place, and 0 for which it did not take place. Then, the linear model coefficients are analyzed to check if the event had an impact on the tracked KPI. This method is simpler than ARIMA while, according to Lagarde, producing more robust estimates of event impact than basic approaches, similar to the before and after comparison taken by Mbugua et al. (1995). However, Wang et al. (2019) criticizes the segmented linear regression approach of Lagarde (2012), mentioning that it will not perform well if the KPI time series exhibits complex behaviors.

Another approach to event impact analysis as referenced by Wang et al. is the usage of change point detection algorithms. Wang et al. (2019) reports that Jarušková (1997) and Adams and MacKay (2007) use such algorithms to detect both the occurrence and timing of an event. However, if the timing of detected event does not coincide with the event of interest, this technique loses its purpose if the goal is to study the impact of a predefined event.

Finally, the most recent approach referenced in this thesis is described at the 2019 18th IEEE International Conference on Machine Learning and Applications by Wang et al. (2019). The authors propose training a ML model on a KPI sequence before the event took place. Subsequently, the KPI can be forecasted beyond the event horizon. Under the assumption that the forecast is highly accurate, the event impact on the KPI can be determined by a comparison between forecasted KPI values and actual KPI values. Wang et al. (2019) claim that this technique outperforms the aforementioned methods by, for example, handling noisy data better, and modeling the dynamics of the data on a more granular level than the methods which rely on modeling aggregated KPI time series. Wang et al. (2019) test this event impact analysis framework in two scenarios with simulated data sets, and three sets of real-life data: they research the impact of equipment maintenance, impact of a traffic accident, and impact of the announcement of new products. These examples are varied, which shows how the statistical methods presented transcend specific industries. This thesis makes use of that fact by applying the idea of Wang et al. to a new industry and extending it, which is outlined in detail in the next chapter.

4 Methodology

4.1 Choosing the relevant KPI

According to Wang et al. (2019), the two major components involved in formally defining an event impact analysis problem are the choice of KPI and an event of interest. For the purposes of this application, the latter is predetermined - the event is given by the partner company. However, the choice of a relevant KPI remains a problem without a single obvious answer. Wang et al. make an observation that the KPI choice requires deep domain knowledge. In this research, qualitative interviews with industry experts within Dataloy Systems and the anonymous customer shipping company as well as the author's previous knowledge gained through professional research and experience with maritime data analytics are applied to specify the relevant indicator.

Qualitative interviews suggest that there exists a prevailing idea within the industry that some shipping KPIs, such as TCE, are heavily dependent on the volatile shipping market and are not an accurate reflection of the schedulers' work, therefore they are not ideal for quantifying the effect of scheduling DSS. To achieve high quality results, the chosen KPI should capture the event effect in a way that is explainable by business logic and agreed on by industry experts. Therefore, in this thesis a choice is made to account for the market effects by selecting a more market independent KPI.

This criterion of a good KPI to measure the success of fleet schedulers is met for example by measuring the time a vessel arrives into the port too late or too early, i.e. the number of days before and after laycan or capturing changes in vessel speed. Researching inefficiencies in planning how fast a shipping vessel should move would be within the domain affected by FAS and to a larger extent independent of market forces, but this kind of data was not available to Dataloy in the time frame required to measure the effects of the customer company purchasing and implementing the software.

Finally, Vessel Weight Utilization, was chosen as an indicator which is an intersection between the set of possible KPIs calculated with available data collected by the customer shipping company, and the set of KPIs which to a large extent reflect the work of the schedulers who use Dataloy's FAS. For the purpose of this research, VWU is defined as the ratio of cargo weight on board of a shipping vessel and the deadweight of the vessel, which is a good proxy for the total cargo carrying capacity. The relevance of this KPI was confirmed by performing qualitative interviews with industry experts within Dataloy Systems and the customer shipping company.

4.2 Data preprocessing

4.2.1 Data sourcing and feature engineering

The data of the anonymous maritime shipping company is sourced from the database of Dataloy Systems AS. Relevant variables to be sourced are chosen based on consultations with maritime industry experts within Dataloy Systems AS as well as the author's qualitative assessment related to previous maritime data analytics experience. The potential predictors which are available and reasonably complete for the anonymous shipping company are qualitatively assessed and screened based on their market independence as well as their potential to be drivers of VWU. This process results in identifying a list of predictors to be used in a later stage of the analysis (see Table 4.1).

In the next step, several data sets with maritime data aggregated on the voyage leg level are sourced, cleaned and merged. The analysis in this research is performed in the R programming language (R Core Team, 2021). R packages *tidyverse* (Wickham et al., 2019) and *lubridate* (Grolemund and Wickham, 2011) are used for handling tabular data. The studied period is limited to 4 years ex-ante the event and 2 years ex-post the event. Additionally, only data from bulk cargo vessels is analyzed. Project carriers were excluded from the study, since in their case it is not relevant to analyze VWU. Similarly, Time Charter voyages are not included in this analysis, because the data generated by those voyages is not influenced by FAS, since they are performed by hired vessels.

Some features are dropped already at this stage due to large amounts of NAs and are not included in Table 4.1. The final data frame consists of the following features:

Table 4.1: Selected variables.

Variable name	**Variable role**	**Description**
voyage_start_date_gmt	time variable	start of each voyage in GMT time zone
leg_id	ID variable	unique identifier of each leg
voyage_reference	ID variable	unique identifier of each voyage
vessel_weight_utilization	dependent variable	studied KPI
leg_no	predictor	number of legs in a voyage
restricting_deadweight	predictor	deadweight of the shipping vessel
from_port	predictor	origin port in a leg
to_port	predictor	destination port in a leg
reason_for_call_from_port	predictor	reason why a vessel was in the origin port, e.g. loading
reason_for_call_to_port	predictor	reason why a vessel was in the destination port, e.g. discharging
vessel_name	predictor	name of a specific vessel
days	predictor	number of days per leg
distance	predictor	distance in miles per leg
speed	predictor	average speed in knots per leg
volume	predictor	volume of cargo per leg
do_consumption_low_sulphur	predictor	consumption of diesel oil low sulphur bunkers
do_consumption_high_sulphur	predictor	consumption of diesel oil high sulphur bunkers
fo_consumption_low_sulphur	predictor	consumption of fuel oil low sulphur bunkers
fo_consumption_high_sulphur	predictor	consumption of fuel oil high sulphur bunkers
miles_ballast	predictor	number of nautical miles in a voyage a vessel travelled without any cargo

Continuation of Table 4.1

Variable name	**Variable role**	**Description**
miles_loaded	predictor	number of nautical miles in a voyage a vessel travelled with cargo
commodity	predictor	type of bulk cargo transported
vessel_type	predictor	type of vessel, e.g. selfdischarger
days_total	predictor	total number of days in a voyage
days_ballast	predictor	number of days in a voyage a vessel travelled without any cargo
days_loaded	predictor	number of days in a voyage a vessel travelled with cargo on board
consumption_leg	predictor	sum of consumption of all types of bunkers per leg
fuel_lowsulphur_used	predictor	1 if fuel oil low sulphur fuel was used, 0 otherwise
fuel_highsulphu_used	predictor	1 if fuel oil high sulphur fuel was used, 0 otherwise
diesel_highsulphur_used	predictor	1 if diesel oil high sulphur fuel was used, 0 otherwise
diesel_lowsulphur_used	predictor	1 if diesel oil low sulphur fuel was used, 0 otherwise
route_unique	predictor	categorical variable signifying the unique route between two ports in a leg, without taking account the direction of travel. E.g. Gdańsk - Helsinki is counted as the same route as Helsinki – Gdańsk
voyage_time_loaded_share	predictor	ratio of time a vessel was loaded in a voyage and time a vessel was ballasting

Continuation of Table 4.1

Variable name	**Variable role**	**Description**
voyage_ballast_distance_share	predictor	ratio of distance a vessel was ballasting in a voyage and the total voyage distance
no_legs	predictor	total number of legs in a given voyage
vessel_time_utilization	predictor	time in a voyage a vessel was loaded divided by the sum of the time the vessel was loaded and ballasting

4.2.2 Handling data quality issues

Because of model specifications, the predictors in the data set used in the future ML model should not have NA values. Features containing large amounts of NA values were discarded at an earlier stage of data preprocessing. Otherwise, rows including the missing data points would have to be removed, which would severely limit the number of observations in the cleaned data set. Those observations could have the ability to add predicting power to the model by supplying information present in other, more complete columns. The leftover selected and engineered features still include some NA values. In their case, filtering is used to remove bad data instances. Removing too many columns with predictors could potentially hamper the predictive power of the model. However, Belgiu and Drăguţ (2016) and Zhou et al. (2016) report that RF algorithms tend to work well with multidimensional data and are not sensitive to multicollinearity. Therefore, if RF is implemented, then including a large number of predictors in a model should not pose a danger to its out-of-sample predictive ability.

Furthermore, the data used is user generated, and therefore contains mistakes and inconsistencies resulting from human error or negligence. For example, 68 observations contained negative number of duration days and negative average speed per voyage leg, and in 35 cases registered departure from a port of origin is an earlier date than the arrival date in the destination port. Such mistakes could happen when a user makes a wrong entry in Dataloy's software. Filtering is used to identify and remove such observations which contain typos and logical mistakes in the data. Additionally, entries which were only test voyages are removed from the analysis. Observations corresponding to legs which lasted more than 50 days are also removed to account for cases where vessel maintenance work was registered as a voyage leg in Dataloy's software. This number was chosen arbitrarily after consulting shipping industry experts from Dataloy and is valid for the case of the specific customer shipping company. The preprocessing steps of filtering types of data relevant for the study as well as cleaning the data set from test entries, mistakes, etc, reduce the number of voyage leg observations from 22,996 to 7,523.

4.3 Applying existing methodology to a new industry

4.3.1 RF Model 1 set up and fitting

Random Forest is the ML technique which is chosen for this analysis for reasons described in Section 4.2.2. In this research, ML models are deployed within the *tidymodels* framework (Kuhn and Wickham, 2020). Furthermore, the preprocessing steps allow for the maritime data to be split in a way suggested by Wang et al. (2019). The chronological split between pre-event training and post-event testing sets is made using the time variable *voyage_ start_ date_ gmt*. This variable is later removed from the data set and not included as a predictor. The logic behind this decision is not to treat the data set as a multivariate time-series, but rather two cross-sectional data sets sourced in two different periods: one in the period before the studied event, and one in the period after the studied event. This is designed to help with highlighting any potential differences between data generated by a company utilizing traditional methods and data generated by the same company using FAS.

Is is important to call attention to the fact that the variables presented in Table 4.1 are not used in the later ML models in the same form. Instead, one hot encoding is used to handle categorical variables. For every unordered factor in factor variables a binary dummy is created. For example, for each reason for call in the destination port, a binary dummy variable is created with the value of 1 if the vessel called into the destination port for a given reason in a given leg and a value of 0 if it did not. One problem with this technique is the high cardinality of the variables. For example, a feature *route_ directional* was initially defined as a categorical variable describing a route between two ports while taking the direction of the route into account, such that Gdańsk - Helsinki would be a different route from Helsinki – Gdańsk. This variable was dropped because the number of individual routes was equal to 93% of the number of observations in general. The number of unique routes still constitutes 80%, which is a large portion of the total number of routes, but it was kept in the model because there was a higher chance that the unique routes could be repeated and used as predictors. The directional routes do not repeat often, since the customer company operates in the spot market. One can speculate that the route variables could have more predictive power if they had lower cardinality.

Subsequent to one hot encoding, the RF model hyperparameters are tuned based on a random grid. Those parameters are *mtry*, which is the number of predictors that are sampled at random with each split, the number of trees contained in the ensemble RF model, and *min_ n*, which signifies the minimum number of data points in a node that are required for the node to be split further (Kuhn and Wickham, 2020). For each of the 10 randomly generated model configurations in the grid, a RF model is fitted on a data set comprised of randomly selected 75% of pre-event training data, and validated against and a set comprised of the remaining 25% of pre-event data. This process is expedited by employing parallel computing using the *parallel* package in R (R Core Team, 2021). Using just two PCU cores instead of one resulted in a 34% execution time saving. Following the repeated model fitting and validation, exact specifications of each model with corresponding RMSE and R-squared metrics are collected. RMSE is chosen as the metric to rate the models, because it penalizes large errors more than small ones, as opposed to alternatives such the MAE. It is also more easily interpretable than another popular metric of MSE, because its value corresponds to the value of the studied KPI itself. For example, RMSE can easily be put in the context of VWU, since an RMSE of 0.4 can be explained as an error of 0.4 percentage points in measuring how filled up a cargo vessel is. Since by its nature the VWU ranges from 0 to almost 100%, the achieved RMSE value is very close to its corresponding normalized RMSE value, which can be used as a good way of comparing ML models. The hyperparameter tuning process is repeated 12 times, each with time with a different approach to one hot encoding and and a different *mtry* range parameter. The *mtry* range can be understood as the lowest and highest allowed number of columns which are sampled at each split when creating the tree models.

As a result of the selection process described above, the best performing RF model is identified and named RF Model 1. This chosen model has the *mtry* parameter of 329, is comprised of 779 trees, has the *min_ n* parameter of 38. The main idea of this analysis is to use RF Model 1 to make a forecast of the studied KPI using only pre-event data for model training and validation. This way, the VWU forecast made for the post-event time period is a fabricated counterfactual reflecting the pre-event way of doing business. In other words, this forecast can be interpreted as a hypothetical "would be" scenario in which the customer shipping company did not implement the software. Unlike the hyperparameter tuning, this process cannot be parallelized because there is only one sequential process in making of the ensemble tree-based model. Parallelization of hyperparameter tuning could be achieved, because different potential models from the random grid could be fitted and validated simultaneously.

Subsequently, a comparison made between this modeled hypothetical scenario VWU and the real shipping data in which Dataloy's FAS software was implemented is designed to capture the effect that FAS had on the studied KPI of the maritime shipping company. According to Wang et al. (2019), this idea is a "systematic and statistically sound way" of answering the question whether a particular event made a statistically significant impact on the operations of a company as defined by a selected KPI. The plot of predictions resulting from this model, as well as other visualizations in this analysis are made with *plotly* (Sievert, 2020). In this study, this comparison is a simple difference between the two vectors, rather then traditional way of calculating a squared difference metric in comparing the prediction to the test set. This way, it is not the absolute distance between the actual KPI values and the fabricated counterfactual KPI is calculated, but the difference including information about the direction of change between the two scenarios. Simply subtracting one vector from the other provides information not only whether software implementation had an impact on the KPI, but also whether that potential impact was negative or positive. This difference, interpreted as the software effect, can be aggregated over the studied post-event time period to produce a mean software effect (see Equation 4.1), which is the answer to the first research question.

The mean software effect is given by the equation:

$$\mu = \sum_{n=1}^{N} (a_n - p_n)/N \tag{4.1}$$

Where μ is the mean software effect, n is each voyage leg in the total number of voyage legs N done by vessels of the customer shipping company over the studied post-event period, a is the actual measured KPI value, and p is the predicted "would be" KPI value. A positive mean software effect means that, on average over the studied period, VWU increased due to implementing FAS by the customer shipping company. Conversely, a negative mean software effect means that the measured cargo tonnage capacity utilization decreased due to the effect of implementing FAS.

Additionally, the numerical outcome of the research question is supplemented by variable importance analysis and a line plot of the factual and counterfactual VWU over time. Variable importance plot showing a sorted table of the most influential features which affect the prediction results is generated using the *vip* package in R (Greenwell and Boehmke, 2020). The impurity method calculated with residual sum of squares is chosen for classifying features importance. It is a standard method of calculating feature importance in regression RF models (Greenwell and Boehmke, 2020). This way, the feature importance score of a given feature is calculated based on the total decrease in node impurities from splitting on that feature, averaged over all trees (Greenwell and Boehmke, 2020).

Besides, it is important to point out that while a certain level of industry knowledge is required to take this approach, for example for feature engineering, the industry-independence of this framework allows for the focus to be on the data science aspect and not the shipping industry aspect. Consequently, it assures that the principles of this event impact analysis method remain the same across different data sets. Therefore, while the overall goal of the study is to unveil the answer to the research question, the described experience of applying this methodology is an interesting insight in itself.

4.3.2 OLS Model 1 as benchmark for RF Model 1

In their study, Wang et al. (2019) propose checking the ML solution against a naive benchmark model. In this research, OLS method is chosen as a way to put the ML model in a wider context. This decision is motivated by the fact that OLS is the simplest linear model which is adequate for regression, while at the same time having a straightforward interpretation.

While RF models can handle multicollinearity and multidimensionality in the data (see Section 4.2.2), the benchmark linear model would be prone to those problems. This would create an issue especially due to the use of the one hot encoding technique which converts several categorical variables into a large number of binary variables. This is mitigated by limiting the number of features used in the OLS model to the top 10 best performing variables of RF Model 1 defined through feature importance analysis (see Figure 5.2). While perfect comparability of the methods cannot be achieved due to aforementioned limitations in the possible number of features, OLS Model 1 is also trained on data split in the same way as RF Model 1 (see Section 4.3.1).

Subsequently, coefficient analysis can provide insights about the significance level of each predictor. More importantly, the RMSE and R-squared values achieved by this model can be compared with the same values obtained by the equivalent ML model. This can constitute a sanity check for this method. Since it can be expected that the much more advanced ML method can outperform a simple linear regression, the OLS model obtaining a larger RMSE than the RF model would be a sign that there was a mistake in the RF model set up process. Moreover, quantifying the difference between the RMSE scores of OLS Model 1 and RF Model 1 can provide information about the extent to which the ML is superior to the linear solution, and presumably confirm that the RF method is indeed the appropriate method for this kind of analysis as compared to a naive benchmark method.

4.4 Modifications to extend existing methodology

The approach of Wang et al. (2019) provides a useful framework which enables a comparison between actual data and the predicted scenario of a shipping company not using the scheduling software. In this study, the insights achieved by implementing this framework involve visualizing the performance of the actual and predicted scenario data over time, as well as calculating a numerical indicator of the change of performance of the KPI between the predicted scenario and the real data. This numerical indicator is a value aggregated over a two-year time period and clearly communicates the averaged-out effect of buying the software on the performance of the shipping KPI.

However, the approach suggested by Wang et al. (2019) has some limitations. The analysis described in sections 4.3.1 performed by deploying RF Model 1 was useful in providing the sufficient statistic of the overall software effect. However, due to the nature of predictions generated by a model minimizing RMSE, it is not possible to derive many insights from comparing the distributions of the predicted KPI time series with the actual KPI time series.

While analyzing the distributions, it can be found that the prediction data is much less variable than the actual data. However, this should not be interpreted as an inherent characteristic of the relationships between the two KPI time series. Instead, this behavior could be an effect of the low bias of the model, yielding any comparison invalid. It is to be expected that a model chosen by minimizing RMSE will be less variable than the actual real-world time series.

In other words, it is not appropriate to compare the distributions of the predicted counterfactual KPI with the real-life KPI, because the two distributions were generated using different processes. The distribution of predicted KPI was generated by RF Model 1, and the distribution of actual KPI was generated by real world market processes with random shocks. Consequently, the predicted distribution in the "would be" scenario is influenced by the model choice, and the actual KPI distribution is not. Therefore, it is not feasible to derive insights from comparing the two distributions.

Those limitations can be addressed and accounted for by making two predictions, one for the scenario of the company acquiring the software and one for the scenario of the software never being implemented. Comparing the two predicted scenarios, instead for one predicted scenario and the actual data, allows for more meaningful distribution comparison and can help to answer the question whether there is any effect in KPI variability which can be attributed to the software use. Using this method, two distributions are more comparable because they were generated using the same method - the same ML model. This mitigates the bias which was present in the case of comparing a model generated distribution with a measured real-world distribution.

4.4.1 Feature engineering and data splitting modifications

This method requires a modification to feature engineering and data splitting. Firstly, a binary variable *soft* is introduced, taking a value of 1 for all observations where the software was in use, and a value 0 for all observations which were gathered in the period when the software was not in use. A similar approach was taken by Lagarde (2012) who used a binary event variable as one of the predictors in a linear model to assess event impact.

Furthermore, instead of splitting the data into training, validation, and test set time-wise, the entirety of available data is randomly split into a 75% training and 25% validation set. This method does not provide an opportunity for an out of sample prediction, or the future "would be" scenario, but it allows for looking into feature statistical significance explored further in Section 4.4.2.

4.4.2 OLS Model 3 as a binary variable significance check

While the initial OLS Model 1 provided a benchmark for the first ML solution, a similar technique can also be used to check whether the introduced binary software variable is statistically significant.

Namely, the OLS model can be used as a significance check by analyzing the p-value of the *soft* variable. Similarly to OLS Model 1, top 10 most influential features from the RF model are selected and an OLS model is trained on the whole data set, with the modification of including *soft* as an additional predictor. The coefficients of the resulting model can be used to derive insights about the significance of the software as a driver of measured potential difference in VWU. The assumption made in this study is that significant linear relationship between the binary software predictor and the independent variable would mean that software use has a quantifiable effect on the company's KPI.

4.4.3 RF model and OLS benchmark model modifications

Given that OLS Model 3 confirms that the binary software variable is statistically significant, a new RF model is fitted on the new training and validation data described in Section 4.4.1, including the modification of changing the software binary variable.

This is different from traditional out-of-sample testing, but in this case the ultimate goal is not to make a future forecast as correct as possible, but to make in-sample forecast scenarios and compare them. Specifically, RF Model 2 uses original data as training and validation, and then recycles the same data to make two different testing sets – one with $soft = 0$ and one with $soft = 1$. This technique allows for generating and comparing counterfactual scenarios just like in the approach taken by Wang et al. (2019), while also allowing for comparing the distributions, accounting for any biases resulting from the model itself. Comparing a model-generated scenario data with real world data leaves room for the set up of the model to influence the comparison results. However, comparing two model-generated scenarios mitigates this influence and theoretically provides more comparable results, assuming that the model is able to replicate the data with reasonably high accuracy.

Using the approach described above, the mean software effect is given as a function of two predictions estimated by RF Model 2:

$$\mu = \sum_{n=1}^{N} (p\prime_n - p\prime\prime_n)/N \tag{4.2}$$

Similarly to Equation 4.1, in Equation 4.2 μ is the mean software effect, n is each voyage leg in the total number of voyage legs N done by vessels of the customer shipping company over the studied post-event period. The key modification is that the relevant statistic includes a difference between $p\prime$, which is a prediction of the KPI in the factual FAS scenario, and $p\prime\prime$, which is a prediction of the KPI in the counterfactual no-FAS scenario. Equivalently to Equation 4.1, a positive mean software effect means that, on average over the studied period, VWU increased due to implementing FAS by the customer shipping company. Conversely, a negative mean software effect means that the measured cargo tonnage capacity utilization decreased due to the effect of implementing FAS.

Furthermore, RF Model 2 can be benchmarked against a naive OLS Model 2, the same way RF Model 1 was benchmarked against a naive OLS Model 1. OLS Model 2 is comparable to RF Model 2 because it uses the same features and data splitting, namely random 75% training and 25% validation sets which together account for all of the data, both pre and post event. This is the main difference between OLS Model 2 and OLS model 3, which is fit on all of the available data. Just like with OLS Model 1, the validation RMSE achieved by OLS Model 2 can be compared to the validation RMSE achieved by RF Model 2 for the same reasons as explained in detail in Section 4.3.2.

4.5 Summary of deployed models

To summarize, this section provides an overview of all five models used in this analysis.

1. RF Model 1 is based on a chronological split of all data between training and testing according to the date of studied event. Random 25% validation set is then extracted from the training set. A prediction is made using the testing set with actual data. Software effect is calculated by subtracting the prediction results from original data.
2. OLS Model 1 is a naive benchmark model for RF Model 1.
3. OLS Model 3 is a significance check for the *soft* variable. It involves fitting an OLS model on all available data.
4. RF Model 2 is based on a random split of all data between 75% training and 25% validation. Two predictions are made using simulated testing sets created for two different scenarios. Software effect is calculated by subtracting no-FAS prediction results from FAS prediction results.
5. OLS Model 2 is a naive benchmark model for RF Model 2.

4.6 Comparing distributions

The second part of the research question concerns the distributions of the studied KPI in the event in which FAS is applied, and in the event in which FAS is not applied. After RF Model 2 is used to produce two predictions, several techniques can be used to compare the distributions. Significantly different distributions of vectors corresponding to the two scenarios would mean that applying FAS did have an effect on the chosen KPI. In this research, R packages *sjmisc* and *BSDA* are used for statistical analysis (Lüdecke, 2018; Arnholt and Evans, 2021).

A simple way to start the comparison is to look at summary statistics describing spread of the distributions. The fact that the standard deviation of the predicted results is much lower than the standard deviation of the actual KPI can be expected since the model is trained to minimize RMSE. This behavior is also the reason for why these distributions should not be directly compared. However, a useful comparison is the one between no-FAS prediction and FAS prediction. The difference of between the standard deviation measures for those distributions is less than 1%, which suggests that there was no observed significant effect on the variability of VWU as predicted by RF Model 2. After an analysis of summary statistics provides some insights, but more advanced methods are used.

4.6.1 Statistical testing

Applying statistical tests to research the similarity and compare the two distributions can be a more advanced, formal way of answering the second part of the research question. A common way to test if two data sets are different is the Chi-square test. However, this test is only applicable to categorical data, and VWU is continuous. One solution would be to bin the VWU vectors and divide them into categories, for example every 10% of the KPI value. However, the result of this analysis could become too dependent on the method of creating the required categories. Instead, this research will make use of the two-sample Kolmogorov-Smirnov test developed by Smirnov in 1939. This test can be used to compare two distributions which exhibit continuous characteristics, like the two VWU scenario projections. The KS test calculates the D-statistic which in this case is understood as the distance between the curve of the empirical distribution function of the no-FAS scenario VWU distribution, and the curve of the empirical distribution function of the FAS scenario VWU distribution. This distance is given alongside the p-value signifying the probability of the result being incorrect. The null hypothesis of the KS test is that the two samples were drawn from the same distribution, which in this use case would mean that the two scenario distributions are the same. If the null hypothesis can be rejected, then the test result supports the alternative hypothesis that there is a statistically significant difference between the two scenarios. The KS test can be sensitive to distribution characteristics like location or shape of the distribution. However, it appears to be more sensitive in detecting differences in means, in comparison to an alternative recurrence plots technique (Wallot and Leonardi, 2018).

The difference in means can be further explored by applying a two-sample z-test, developed over the years since the 18th century, but described in detail by Lehmann et al. (2005). This test is used to determine whether the means of the two vectors are different from each other. Its alternative hypothesis states that the true difference in means is not equal to 0. If means are indeed different, this would provide evidence supporting the hypothesis that FAS did affect VWU of the customer company. This is a parametric test, which assumes that the two samples are normally distributed. For this reason, each vector is scaled to bring the distribution closer to a normal distribution. However, the distributions are not centered. That would mean performing a full Z-score standardization, essentially automatically ensuring an inability to reject the null hypothesis that the true difference in means is equal to 0, since both the means would be artificially transformed to 0. I choose to implement this test with several standard confidence intervals, like 0.99, 0.95, and 0.90.

4.6.2 Visual analysis

Z-test and KS test can be good ways to quantify the difference between the distributions, but statistical testing should not be used on its own. Some insights can become apparent from simply inspecting the distributions visually. Hence the need for histograms and density plots.

Histograms serve to visualize the number of voyage legs which were predicted to fall into each of the predetermined ranges of VWU. For ease of visual interpretation, the histograms are created to contain 40 bins evenly spaced between 0% and 100% of the KPI.

Subsequently, kernel density estimation is used to create density plots. As reported by El Machkouri (2011), this technique was independently invented by Rosenblatt (1956) and Parzen (1962). A kernel, defined as an even, non-negative, real-valued function is appropriate for visualizing vessel tonnage capacity utilization, because it is a continuous PDF with an area under the curve equal to 1. Hence, it is appropriate for visualizing the continuous KPIs. A kernel function is created at every data point, and the average of all kernels is interpreted as the approximation of the PDF for the researched KPI. Kernel density estimation is a non-parametric, or independent of assuming any underlying distribution method. An important parameter in creating a kernel density estimate is the bandwidth (Trosset, 2009). The bandwidth is the parameter determining how many data points should be included in each of the kernels. The larger the bandwidth, the smoother the graph, therefore for a relatively large data set of tightly packed observations a smaller bandwidth would be more appropriate, since it places more weight on the data point, and not the neighboring data points in each of the kernels. In this study I decide to let the function *density()* from R package *stats* choose the bandwidth parameter automatically (R Core Team, 2021). The result is a smooth plot which can be interpreted similarly to a histogram but captures the continuity of the projected KPIs.

5 Results

The results of the analysis provide answers to both parts of the research question.

5.1 Research Question 1

The first part of the research question concerns the overall effect of the researched software implementation event on VWU, which is vessel cargo weight capacity utilization. To find the answer two ML models were deployed. This answer is supplemented by results from three linear models. This section describes results obtained by using the five models in relation to the first research question.

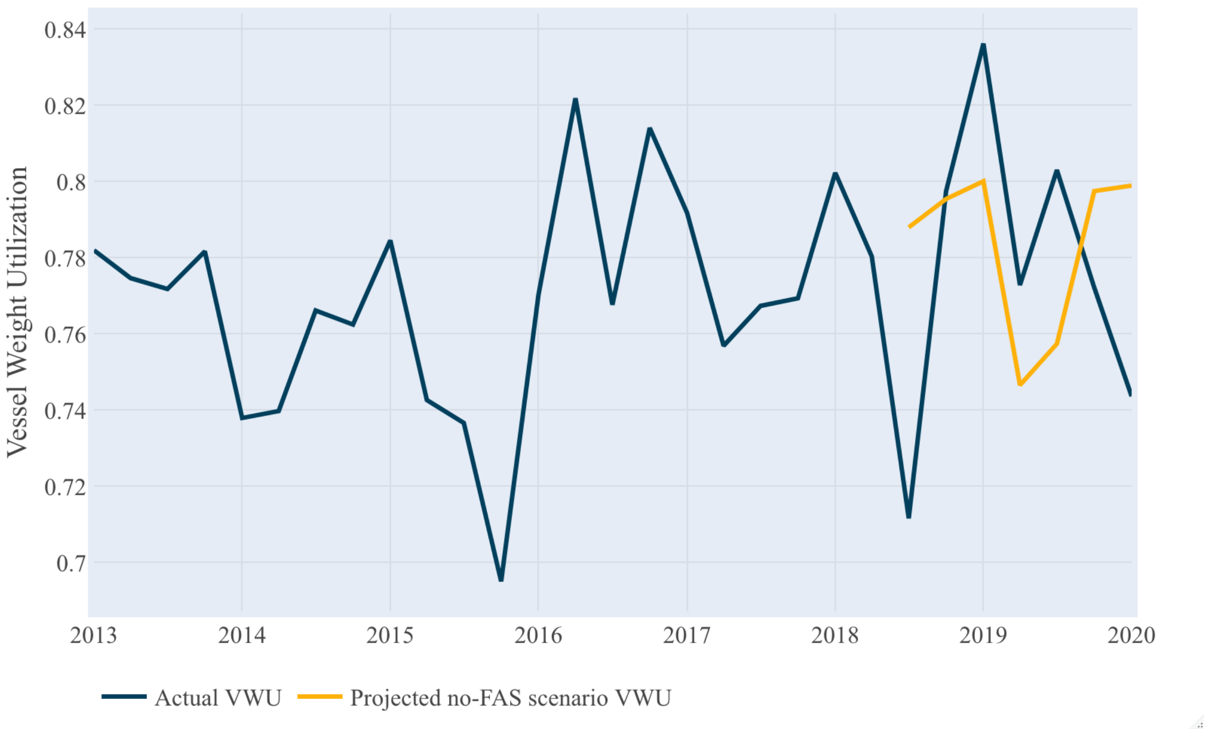
5.1.1 RF model 1 - forecast against actual data

Firstly, 12 different RF models with different model specifications and feature selections were deployed. Subsequently, the validation set RMSE data and the mean and median effect of the software over the test set period information was collected and the relationship between those two values was analyzed. There appears to be an inverse relationship between RMSE values and software effect. Less complex models suggested a slight positive mean effect of 0.02 percentage points more utilized vessel cargo weight capacity. The more exact the model, the more the mean effect converges to 0. The best performing model with 0.125 RMSE resulted in an estimated mean effect of FAS implementation equal to -0.0038 and the estimated median effect equal to 0.0264. The distribution of the estimated software effect ranged from –0.8423 to 0.7408, with the IQR showing that the length of the middle 50% of the interval of space resulting from the prediction was equal to 0.1530.

Furthermore, the analysis provides information about the R-squared value of the models. Saunders et al. (2012) conclude that “unfortunately, there are no set criteria as to what universally represents a “good” R-squared value, so the only way of assessing such a statistic is via comparison with another predictive model.” Following this logic, several models with different RMSE and R-squared scored are compared in this analysis. It can be concluded that as RMSE scores decrease, R-squared values increase. In the best model, 71.1% of the variance in the dependent variable is explained by the fitted model relative to the mean of the dependent variable.

Figure 5.1 presents the actual VWU and the projected "would be" VWU over time aggregated on the quarter level. Lower aggregation level would result in a graph which would be confusing to interpret visually. While a prediction was made for each leg of each voyage within the studied period, which amounts to multiple predictions per day, the predictions are visualized as a time series in an aggregated format for increased interpretability.[1]

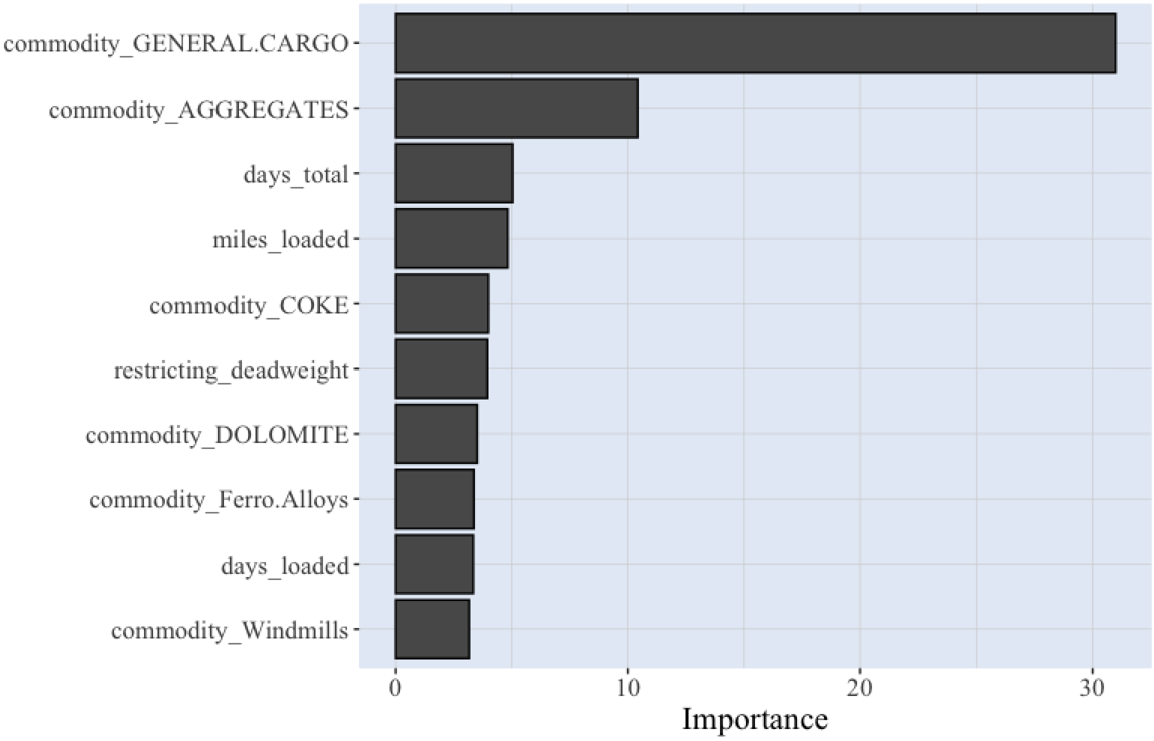
Figure 5.1: Counterfactual scenario VWU predicted by RF Model 1



RF Model 1 used to generate the set of predictions presented in Figure 5.1 was driven by a number of features. The features which influenced the predictions the most are displayed in Figure 5.2:

[1] For the same reason of increased interpretability, colors on the graphs throughout this paper are kept consistent. On all graphs, blue corresponds to actual data, orange corresponds to no-FAS scenario prediction, and green corresponds to FAS-scenario prediction. The blue - orange spectrum is chosen because it is generally more convenient for people with colorblindness than other color palettes. The exact shade of green was chosen to complete a visually equidistant color palette - all colors look equally different and are easiest to tell apart (Learn UI Design, 2022).

Figure 5.2: Variable importance plot of RF Model 1



5.1.2 OLS Model 1 - naive benchmark validation RMSE check for RF Model 1

The software implementation effect results were achieved by applying the framework for event impact analysis described by Wang et al. (2019). The authors also suggest checking the ML model against a naive benchmark. In this paper I check the RMSE of the best performing RF model against a simple OLS model using the top performing RF features. For comparability, the benchmark OLS model was fitted on the pre-software implementation training set of the chronological data split and checked against a random 25% validation set in the same way as in the RF model.

The achieved validation set RMSE was 0.205, which is considerably higher than the RMSE of 0.125 achieved by the RF model. When it comes to coefficient analysis, variables *commodity_ GENERAL CARGO*[2], *commodity_ AGGREGATES*, *commodity_ COKE*, *restricting_ deadweight*, *commodity_ DOLOMITE commodity_ Ferro Alloys*, and *commodity_ Windmills* were statistically significant with p-values close to 0.

[2] Some of the explanatory variables were automatically generated through one hot encoding

Variables *miles_ loaded* and *days_ total* were statistically significant with the p-values of less than 0.001. Variable *days_ loaded* was not statistically significant. The obtained R-square value signifies that the benchmark model explains 37.6% of the variance in the dependent variable, as opposed to the R-squared of 71.1% of the RF model. The model was significant with the p-value less than 2.2×10^{-16}, or close to 0.

5.1.3 OLS Model 3 - software effect statistical significance check

The result from RF Model 1 shows that, on average, implementing the fleet allocation and scheduling software did not affect the KPI in the measured period. Additionally, OLS Model 1 achieved much lower RMSE than RF Model 1, which can be understood as the RF model passing a sanity check. Furthermore, the extent of the difference between the RMSE measures of the two models suggests that the RF technique is appropriate for this research as compared to a benchmark model. This approach of testing a ML model against a naive benchmark was suggested by Wang et al. (2019). However, another method of determining whether an event can possible have an impact on a measured KPI as proposed in this master's thesis involves checking the statistical significance of the effect of the software by introducing a software use binary variable to the model.

The OLS Model 3 was fitted on all the available voyage leg maritime data within the studied period, encompassing the periods before and after FAS software purchase and application. This OLS model regressed VWU on the binary software variable as well as the top 10 most important features. The coefficients of the resulting model were analyzed to answer this part of the research question. The binary software variable was found to be statistically significant with p-value less than 0.05. Out of the other 10 independent variables chosen based on their performance in RF Model 1, only the *days_ loaded* variable was found not to be statistically significant. The remaining nine predictors were significant with p-values less than 0.001. The R-squared of this model was found to be 32%, which is comparable to the 36% achieved by OLS Model 1. The p-value of Model 3 was found to be less than 2.2×10^{-16}. This p-value close to 0, in conjunction with the p-value of the software variable, means that the effect of the *soft* variable is statistically significant.

Furthermore, this model was used to make a prediction for each of the two scenarios corresponding to the two simulated data test sets. Those predictions resulted in the estimated software mean effect of –0.0173 percentage points of VWU.

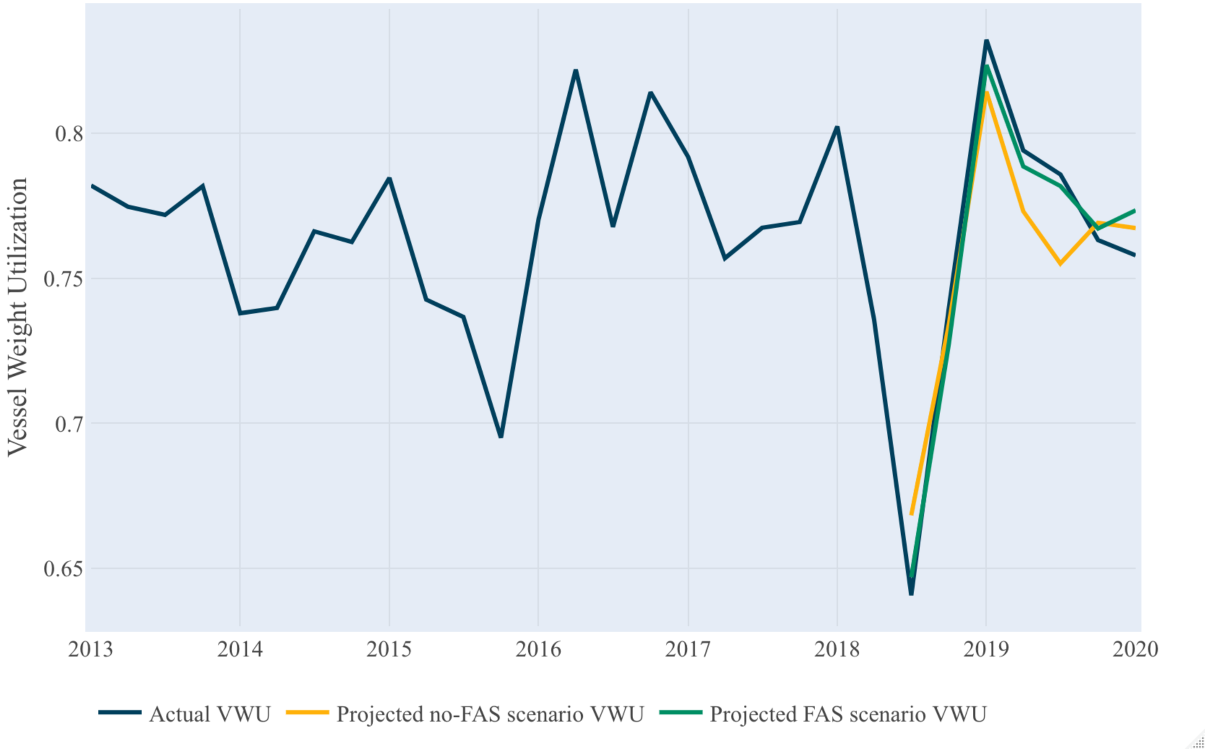
5.1.4 RF model 2 - forecasting and comparing two scenarios

The benchmark OLS Model 3 analysis results suggest that using a binary software variable as a predictor of vessel cargo weight utilization percentage is statistically significant. However, results from a validation RMSE check performed in the analysis of OLS Model 1 show that a linear model is greatly outperformed by an RF model for the purpose of this analysis. Therefore, a second RF model was created as a further novel extension of event impact analysis methodology proposed by Wang et al. (2019).

RF Model 2 was trained and validated on all the available data both ex-ante and ex-post software purchase, and subsequently used to generate two projections: one for the scenario with $soft = 0$ and one with $soft = 1$. The model used the same independent variables as OLS Model 3.

While RF Model 2 was trained on thousands of observations of many variables of maritime data, it is the most insightful to look at the two predictions visualized in a two dimensional space. Figure 5.3 displays factual quarterly VWU values for the whole fleet, a prediction of those values in a factual scenario, and a prediction of the same values in a counterfactual scenario.

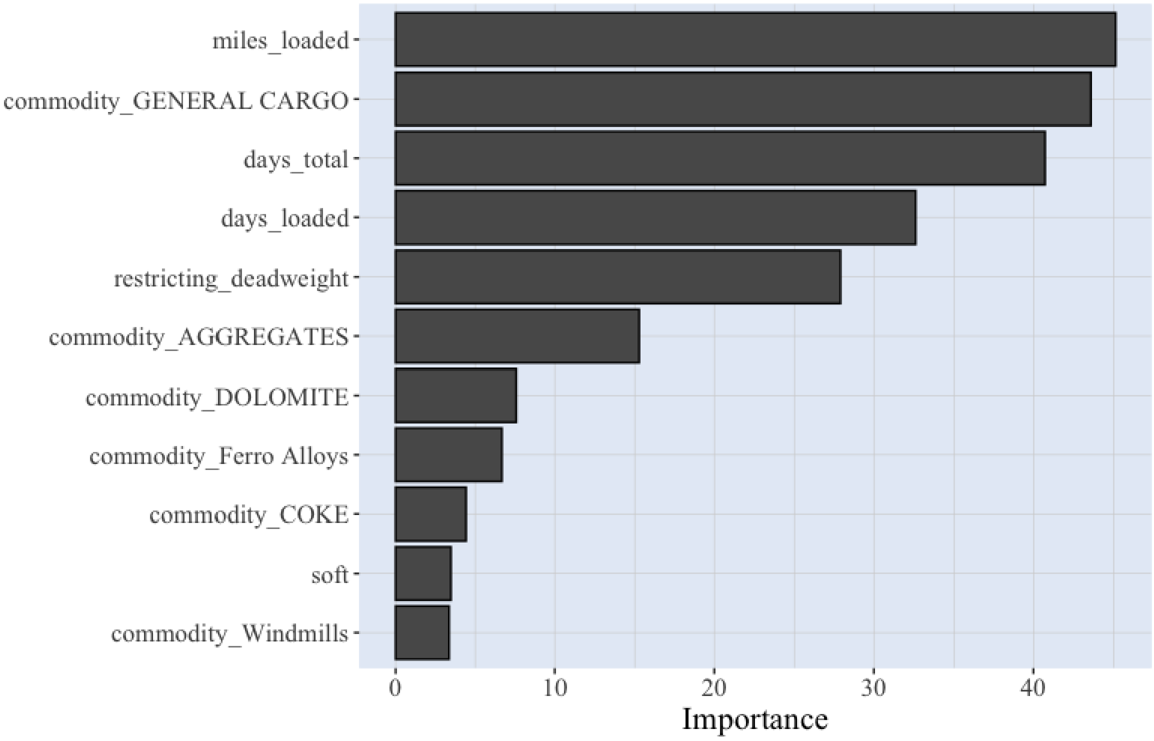
Figure 5.3: Counterfactual scenario and factual scenario VWU predicted by RF Model 2



Similarly to the original RF approach applied before in RF Model 1, the difference between two ML predictions estimated with RF Model 2 which can be interpreted as the software use effect was calculated to be close to 0. The mean software effect over a two-year period after purchasing FAS by the customer company was estimated to be 0.002, while the median was close to 0. The RMSE of RF Model 2 was 0.149, which is lower than the first RF model, but higher than the linear benchmark. RF Model 2 also achieved a worse than RF Model 1 R-squared of 55.6%. The distribution of the estimated software effect ranged from −0.1738 to 0.3352, which is a smaller difference than the previous ML model. Similarly, the IQR of the software effect estimated by RF Model 2 was also smaller, being equal to 0.0024.

Figure 5.4 provides information about the most influential features of RF Model 2. Those are the features which affected the predictions visualized in Figure 5.3.

Figure 5.4: Variable importance plot of RF Model 2



5.1.5 OLS Model 2 - naive OLS benchmark validation RMSE check for RF model 2

The benchmark OLS model fitted on randomly selected observations consisting of 75% of the original data and checked against a validation set of the remaining 25% provides a reference point for RF Model 2 when it comes to validation set RMSE. The resulting RMSE value for the benchmark OLS model was 0.188, which is considerably higher than the RMSE of 0.149 achieved by the RF model. When it comes to coefficient analysis, the binary software variable and the *days_ total* variable were significant with the p-value lower than 0.01. One variable, *days_ loaded*, was not statistically significant. The remaining 8 out of the 11 dependent variables were significant with their p-values close to 0. The obtained R-square value signifies that the benchmark model explains 34% of the variance in the dependent variable, as opposed to the R-squared of 55.6% of the corresponding RF model. The model was significant with the p-value less than 2.2×10^{-16}, or close to 0.

5.2 Research Question 2

The second part of the research question concerns the effect that the software had on the distribution of the measured KPI. As shown in Section 5.1.2, the KPI prediction in the "would be" scenario is a relatively good approximation of what would happen in the real world if the shipping company never implemented FAS. However, as described in Section 4.4, because the real world KPI and the "would be" scenario KPI were generated using different processes, it was not appropriate to compare their distributions. Instead, it was much more adequate to compare the distributions of two predictions generated using the same model, as described in Section 4.4.3. This section describes the results of this comparison.

One way to test whether the studied event had any non-zero impact on the measured KPI was to check whether the FAS scenario prediction and the no-FAS scenario prediction have significantly different distributions. A two-sample KS test was performed on the two vectors. The null hypothesis in this test states that the VWU prediction in the no-FAS scenario and the VWU prediction in the FAS scenario have the same distribution. The D test statistic of the two-sample KS test, which measures the maximum absolute distance of the empirical cumulative distribution functions, was calculated to be 0.0919. A low p-value of 0.0042 provides enough evidence to decisively reject the null hypothesis at 1% significance level. Therefore, the alternative hypothesis that the two VWU predictions have varying distributions holds. However, from this test results it is not known whether the small, but statistically significant difference comes from varying shapes of the distributions, varying means, or other factors.

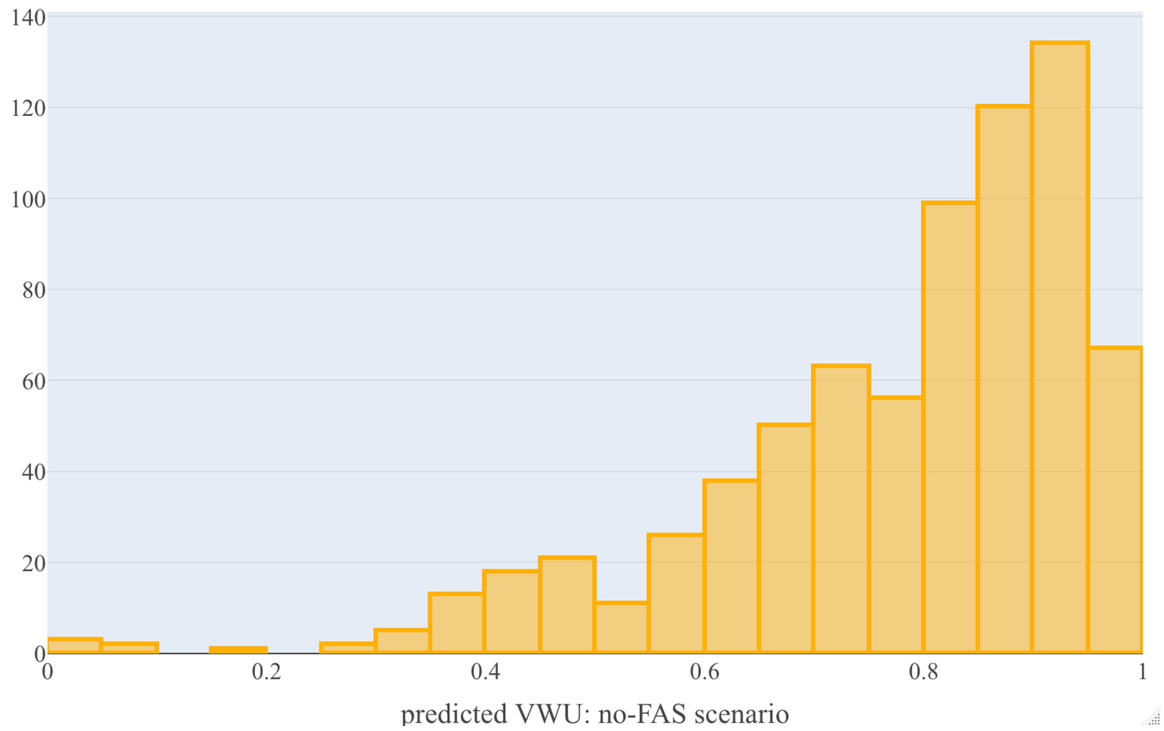
Subsequently, the distributions were further compared using a two-sample z-test test. The z statistic was calculated to be equal to $z = 0.0928$. That low z statistic could be interpreted as the two distributions being the same. However, an important value in the test results is the p-value which was calculated to be 0.926. A high p-value does not allow for rejecting the null hypothesis that the true difference in means is equal to 0.

Thus, statistical testing revealed that the projected KPI distributions are not exactly the same, but that when averaged over time they have the same means.

Finally, the distributions were visually inspected to derive more insights.

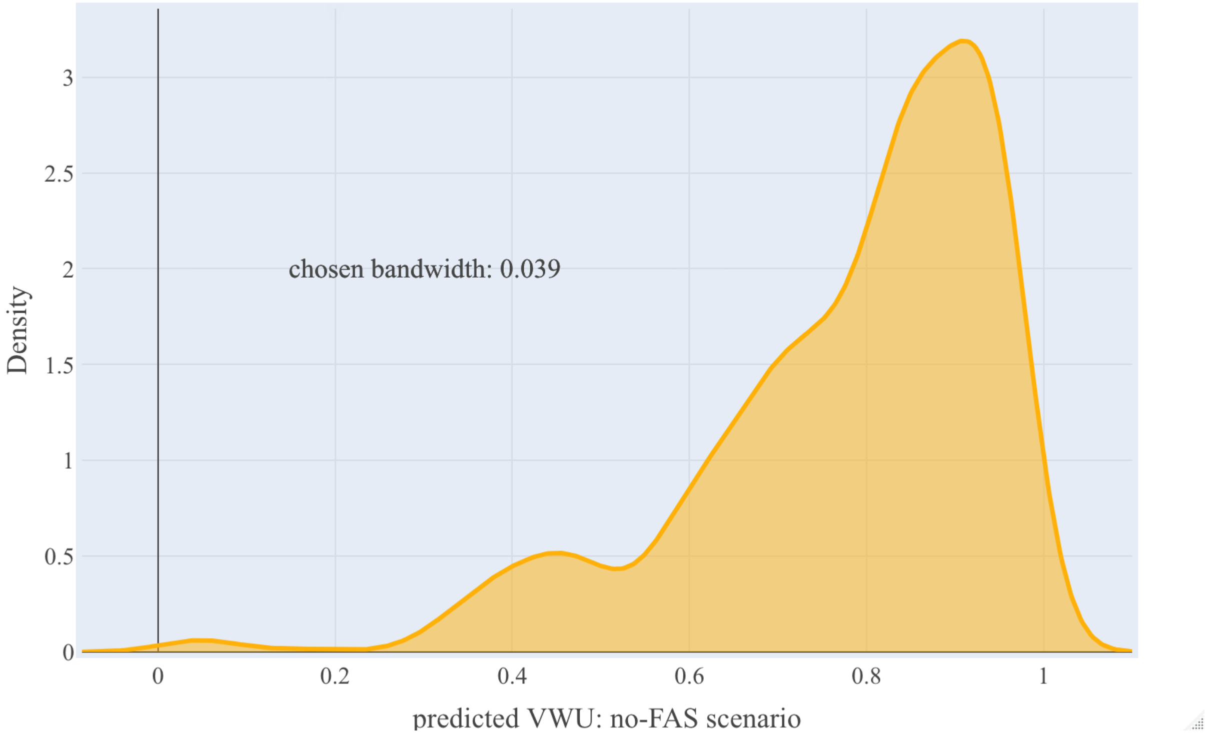
5.2.1 No-FAS scenario prediction distribution

Figure 5.5: Distribution of no-FAS prediction results



As visible in Figure 5.5, in a scenario in which FAS was never implemented the distribution of VWU prediction is bimodal and negatively skewed. The predictions estimated for the 2-year test period ranged from 0.0689 to 0.9753, while the IQR of the distribution of no-FAS prediction results was 0.17543. The average voyage leg over this time period was predicted to be done by a vessel that was 78.02% filled up by weight. As most of the outliers are towards the lower end of the VWU KPI, a typical, median voyage leg was predicted to be done by a vessel that had 81.67% weight capacity utilization. The standard deviation of the predictions was calculated to be 0.1561, which is significantly lower than the standard deviation of the real test data in this period which was 0.2216.

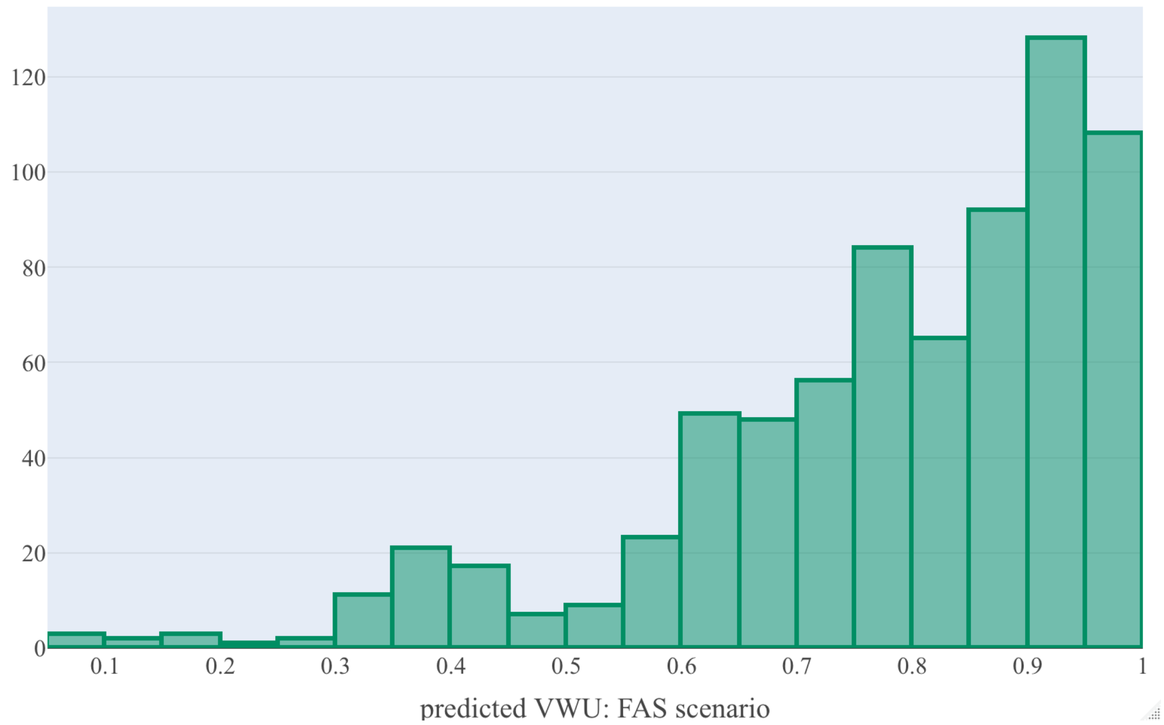
Figure 5.6: Kernel density of no-FAS prediction results



A kernel density plot presented in Figure 5.6 can further validate insights gathered about the distribution of the predicted KPI. The chosen bandwidth for the density plot was 0.039. The density plot provides a continuous visualization of the distribution and confirms the insights about its shape characteristics derived from visual analysis of Figure 5.5.

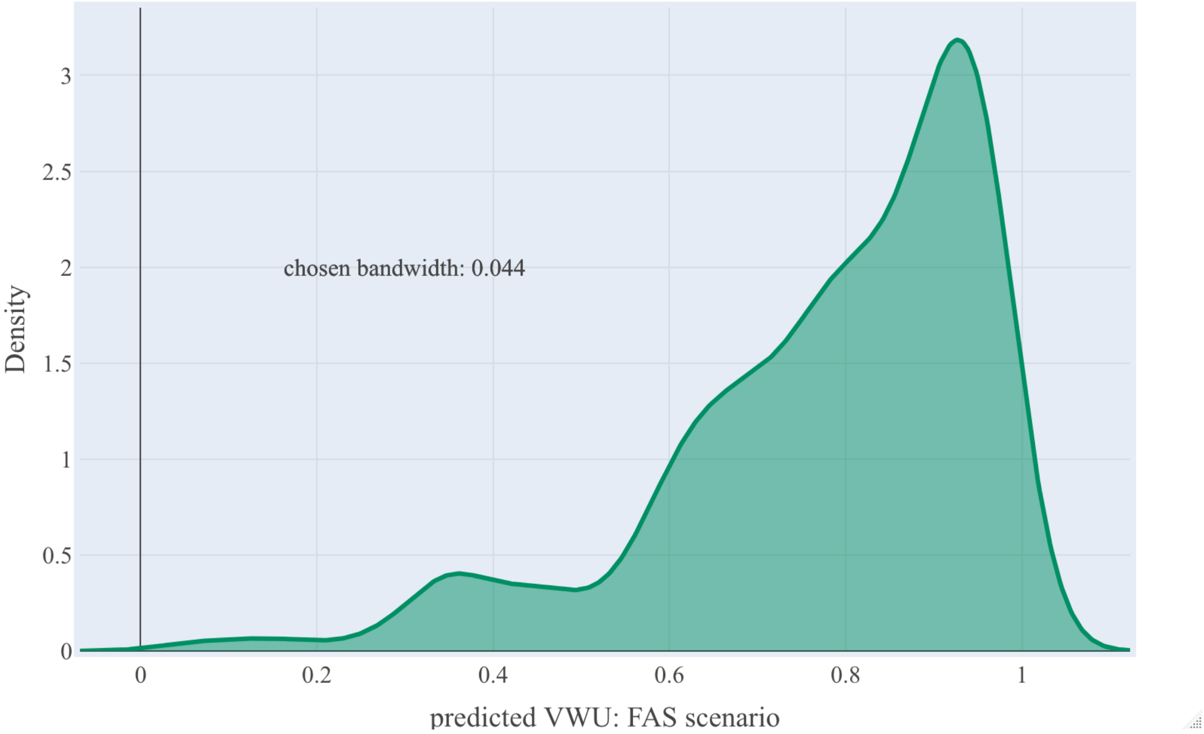
5.2.2 FAS scenario prediction distribution

Figure 5.7: Distribution of FAS prediction results



A visual comparison of Figures 5.5 and 5.7 yields an insight that in the FAS scenario the distribution of RF Model 2 prediction results has a similar shape as the distribution in the no-FAS scenario. The same can be concluded when comparing Figures 5.6 and 5.8. In the FAS scenario distribution, the estimated prediction ranged from 0.1612 to 0.9789 within the test period. This is a slightly smaller range than in the no-FAS scenario. On the other hand, the IQR for the second scenario prediction was equal to 1.9999, which is more varied than in the first scenario prediction. The vessel weight capacity utilization over the testing period for an average voyage leg was estimated to be 78.02%, which is the same as in the first scenario. Similarly, in the FAS scenario the typical vessel weight capacity utilization over the testing period was calculated to be 81%, which is very close to the value of 81.67% predicted in the case of no-FAS scenario. The standard deviation of the predictions in this distribution was calculated to be 0.1549, which is significantly lower than the standard deviation of the real test data in this period, but very close to the standard deviation calculated for the no-FAS scenario.

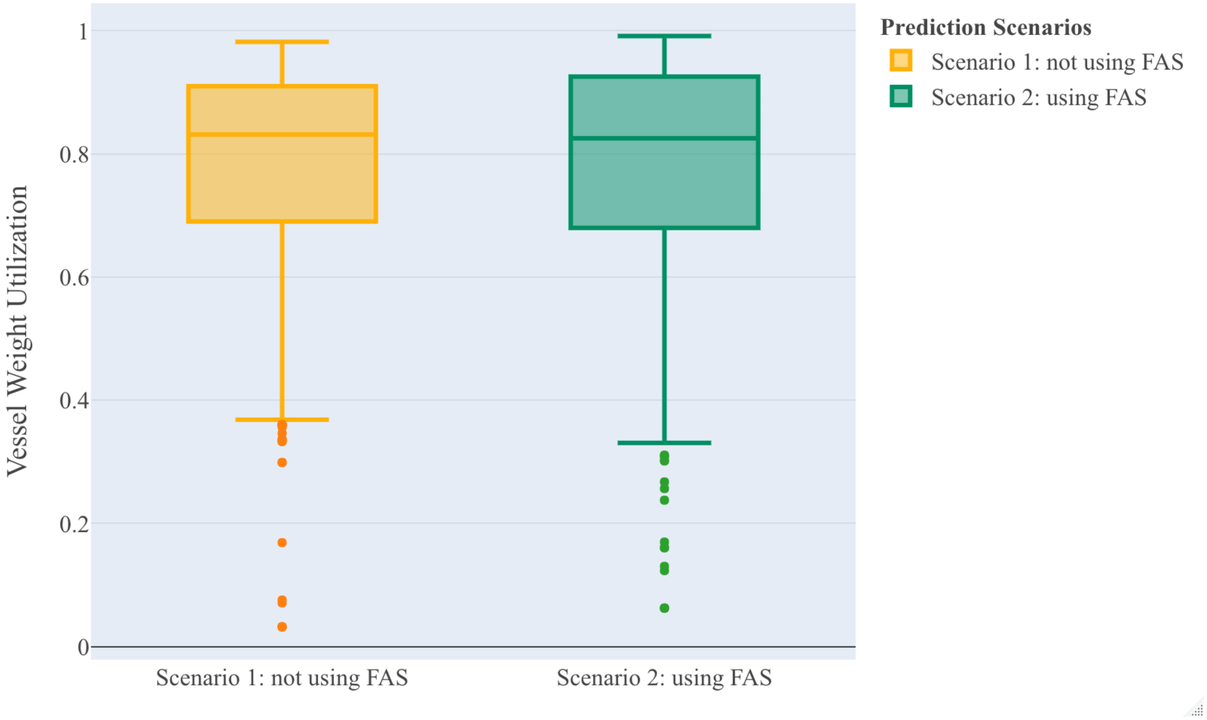
Figure 5.8: Kernel density of FAS prediction results



The chosen bandwidth for the kernel density plot in Figure 5.8 in the no-FAS scenario was 0.044. The two-sample KS test and two-sample the z-test results suggest that the two prediction distributions are different, but it cannot be rejected that their means are the same. However, visual inspection of both histograms and density plots reveals that the distributions themselves look very similar.

The result that while the variance of the two distributions can be different, the distributions themselves look similar is further confirmed by a visual assessment of scenario prediction box plots in Figure 5.9.

Figure 5.9: Box plots of scenario prediction distributions



Overall, a conclusion can be reached that implementation of FAS by the customer company had a minor, but statistically significant effect on the distributions of VWU but did not make a meaningful difference when it comes to the VWU value averaged over the two-year period ex-post the software purchase.

6 Discussion

This section of the paper provides commentary and interpretation of insights described in the Results section, as well connects the described results to their potential real-world applications. Additionally, the contribution made by this thesis is presented. Finally, research limitations are discussed, and possible further research directions are proposed. For the purpose of clarity, research question discussion in this section follows the same narrative outline as the Results section.

6.1 Research Question 1 discussion

The first research question relates to the overall effect of the software on the tracked shipping KPI. I answer it by performing two analyses using ML models, and one significance check by deploying a linear model.

6.1.1 RF Model 1 - forecasting a counterfactual scenario

A key measure in this study is the forecast of how the KPI would behave if software was not implemented. This value should be viewed in conjunction with the actual KPI data, and the difference between the actual and predicted values constitutes the effect of the software. Figure 5.1 shows that the prediction line remains below the actual VWU values for most of the time. However, in a smaller number of points the prediction values are considerably higher than actual values. This interesting behavior visible in the graph is a graphical reflection of the analysis result which shows that the median effect seems to converge around a slightly positive value of 0.02 percentage points difference, while the mean effect is a number close to 0. This could suggest, that for most voyage legs the vessel cargo tonnage capacity was utilized in 0.02 percentage points more than it would be if Dataloy's customer did not implement FAS. However, given the RMSE it is not possible to reject that this result is not due to randomness in the data. In other words, the model suggests that, on average for all voyage legs, FAS had no effect on the utilization of vessel cargo weight capacity. It can be expected that the true effect is between –0.13 and 0.13 percentage points. Correspondingly, the model suggests that during a typical voyage leg the vessel cargo weight capacity was utilized 0.02 percentage points more than it would be if FAS was not used. The true effect on median effect can be expected to be between –0.11 and 0.15 percentage points.

While simpler models suggest a slight positive impact on the measured KPI, more robust models with lower RMSE show no difference between actual data and the theoretical scenario of a shipping company never implementing the software solution. The interpretation of this result is that the shipping company's performance as measured by the vessel tonnage capacity utilization did not decrease with the switch to the new software. Therefore, in light of the result achieved by implementing the methodology of Wang et al. (2019) to a new industry, it can be concluded that implementing Dataloy's FAS by its customer shipping company did not negatively affect the company's operational efficiency as measured by vessel weight capacity utilization. Furthermore, by extension, available data suggests that switching to FAS software does not pose a significant operational risk to the partner shipping company.

This result can be viewed as positive both for the shipping company, and for the software company developing the product. It would be inadequate to expect a significant improvement in KPIs related to mathematical optimization of the schedule, such as the vessel weight capacity utilization. It is important to keep in mind that within the studied period, FAS did not yet implement fleet plan and scheduling mathematical optimization algorithms. During that period, the focus and the biggest underlying advantage of FAS was to streamline and systematize the work of human schedulers, as well as provide the customer companies with a scalable solution with the goal to make seamless switch to a solution with algorithmic optimization recommendation mechanisms in the foreseeable future. Due to the current UX benefits as well as the expected future added value of the mathematical optimization benefits, it is a satisfying result for a DSS software solution which is proven to not pose a greater risk than the traditional methods, but rather provide an expectation of being "future proof". For the shipping company, the described result means that the operational risk is not significant and should not be a reason for concern. For the software company, the outcome of this analysis provides feedback that for the time being their software DSS solution is able to replicate the performance of methods relying solely on human skill and experience. This provides evidence that Dataloy AS is in the right place to begin implementation of more advanced mathematical optimization algorithms. Interestingly, during the feature importance analysis *restricting_ deadweight* was chosen as one of the most important predictors of VWU. In the perfect world, vessel size should not determine the degree to which the vessel is filled up with cargo, since the vessel should be chosen to fit the cargo. However, it is possible to predict how filled up a bulk carrier will be based on its size. This implies that vessels are not always perfectly matched with the amount of the cargo. This could be either due to vessels of more adequate size not being available in the portfolio of the shipping company, or due to a tendency to select vessels that are not the optimal size for a given cargo.

6.1.2 OLS Model 1 - naive benchmark validation RMSE check for RF model 1

The results from the naive benchmark check of the first ML model suggest that the chosen RF approach is a correct technique for performing this analysis. By extension, this model performance check also suggests that the chosen statistic calculated as the difference between real-world and prediction data is sufficient to derive meaningful insights about the operational performance of the studied shipping company. Overall, the results from OLS Model 1 provide evidence supporting the reliability of RF Model 1 results and their business-context interpretation.

6.1.3 OLS Model 3 - software effect statistical significance check

As described in the methodology chapter, the OLS model explanatory variables were selected based on the top performing features in the RF model. Hence, it is not surprising that most of the features are statistically significant with p-values close to 0. However, what is key for this part of the analysis is the significance figure for the software dummy variable. The result from OLS Model 3 shows that this variable is statistically significant with p-value less than 0.001. This means that the extension of event impact analysis methodology proposed in this master's thesis passes the significance check designed specifically for this study. The decision to introduce a binary software variable and in the later stages of the research project use it to model scenario VWU holds its ground due to this variable being statistically significant in OLS Model 3.

While this linear model provided a variable significance check performed by analyzing the coefficients significance scores, it cannot provide any information on the validation set RMSE. This is because the model was fitted on all available data without reserving any observations for a validation set. For this reason, the predictions estimated using this model are not comparable with the other models and are not as trustworthy. This method cannot be treated as a possible accurate prediction model, but rather only serve a purpose of checking variable significance levels.

6.1.4 RF model 2 - forecasting and comparing two scenarios

The methodology introduced in this thesis involves a comparison of two scenario predictions. While this approach is still designed to provide an answer to Research Question 1, the chosen relevant statistic is calculated differently. They key value of average software effect over a 2-year period after implementing FAS is calculated as a difference between the predicted VWU in a scenario in which the customer company does not use FAS, and the predicted VWU in a scenario in which the customer company does use FAS. As seen in Figure 5.3, the predicted time series in a FAS scenario seems to follow the actual data closer than the non-FAS scenario. This suggests that RF Model 2 was able to adequately replicate the behavior of the actual VWU time series.

However, when averaging out the years, the final outcome shows that there was no significant difference between the outcomes in the two simulated scenarios. The result of this part of the analysis confirms the result produced by RF Model 1. Answering Research Question 1 using two different methods, one inspired by Wang et al. (2019) and one original, essentially provides a cross validation of the achieved study results. Since the result produced with RF Model 2 is the same as the result produced with RF Model 1, the same interpretation of study result holds from the perspective of the software company, as wear as the shipping company purchasing the software.

Moreover, further comments and conclusions can be made about the software effect in the context of organizational knowledge and time efficiency in the operational processes of the customer shipping company.

Firstly, maritime shipping companies rely on the collective knowledge and skills of their schedulers. The traditional manual ways of making fleet plans could correspond to a risk that the quality of such fleet plan would decrease if the experienced professionals were to leave the company. The result of this analysis suggests that implementing a DSS software for fleet allocation and scheduling can result in fleet plans of the same quality as measured by the tracked KPI. However, a hidden benefit of implementing this solution is that the risks related to producing sub-optimal plans are spread out over both the DSS as well as the expertise of human schedulers. Those risks are no longer a function of just one variable in the equation – the human element. A synergy achieved by combining the skill of shipping professionals with the existing and potential future benefits of scheduling software could potentially contribute to current way of doing business being retained by an organization even in the case that some of the schedulers were to retire or for any other reasons part ways with a shipping company.

Secondly, since the results show that in the context of VWU a scenario in which the DSS solution is used, and the company spreads out the responsibility for a fleet schedule to both human intuition and the software, can be just as good as a scenario in which all of the responsibility lies on the shoulders of schedulers, a potential benefit of saving time can be expected. Under the assumption that the improved UX can make the allocation and scheduling process quicker and more transparent, the reclaimed time of shipping professionals can be applied to other areas withing the organization. One can hypothesize that streamlining the fleet allocation and scheduling process can be used to improve overall organizational efficiency related to people within the company.

Additionally, another interpretation of the results, this time from the perspective of the software company, is that it can utilize the insights from this research to facilitate building of trust with existing and potential new customers. This analysis concludes that there is no significant operational risk in switching to a digital DSS solution. This knowledge can be used as basis for forming long-lasting professional relationships between maritime software companies and their maritime shipping customers.

6.1.5 OLS Model 2 - naive OLS benchmark validation RMSE check for RF model 2

A novel approach proposed in this master thesis was to analyze the impact of applying new maritime software by training and validating a ML algorithm on all the available data, both ex-ante and ex-post software purchase. Naturally, the quality of this approach had to be separately tested. Key insights were derived from the analysis of the regression beta coefficients and their statistical significance. The analysis of this benchmark OLS model shows that the RF algorithm makes sense as a method of choice for this research also within the new framework involving a comparison of two ML predictions.

Additionally, OLS Model 2 can be thought of as a robustness check of OLS Model 1. Since the second linear model was created using the same features as the first linear model, plus the binary soft variable, it can be used to provide insights about that variable. This result further confirms the validity of using software use as a relevant predictor of measured KPI.

6.2 Research Question 2 Discussion

The second research question relates to the effect of the software on distribution of the tracked shipping KPI. I answer it by performing statistical testing and visual analysis of the distributions generated by RF Model 2 for two alternative scenarios: one where the customer company never implements FAS, and one where it does implement FAS.

6.2.1 Statistical testing

The KS test confirmed that the distributions of the two KPI predictions are different. The test statistic can be interpreted that the probability of the difference between the distributions being larger than D = 0.09 is smaller than 0.4%. However, this test does not show where the difference comes from – the location such as mean or median, or the shape of the distributions, such as skewness or kurtosis.

Ultimately, the shipping company's success in the context of the utilization of vessel cargo carrying capacity would be mostly reflected in the mean. By extension, this is the statistic which would also be key from the perspective of the software company, which can be assumed to want to maximize the customer's success. The results of the two-sample z-test suggest that it is not possible to prove that the no-FAS and the FAS distributions means are different. Indeed, the descriptive statistics results of the no-FAS VWU distribution and FAS VWU distribution suggests that the two means are very similar, both close to 78%. Those results can be interpreted to mean that while the VWU distributions are not the same, the differences do not come from the average VWU values aggregated over time.

6.2.2 Visual analysis

An analysis of visualizations of the VWU scenario predictions revealed that the shape of their distribution is similar, despite the quantitative statistical testing showing that the distributions are not exactly the same. This holds both for the shape of the histograms, as well as the kernel density plots. For the stakeholders, this means that switching from more manual methods to a fleet allocation and scheduling software does not result in an increased operational risk also from the perspective of the measured KPI distributions.

6.3 Contribution

My initial approach is based on the approach by Wang et al. (2019). They test the framework on two simulated data sets and a real-life application. My approach is more applied, since I strive to solve a concrete business problem. The contribution made by this thesis involves applying the described methodology to a new industry of maritime transportation. This application is useful since digitalization and use of data to make business decisions has lagged behind in maritime shipping compared to air cargo or truck transport. In addition, this thesis makes a contribution by extending a framework previously described in literature.

Furthermore, the literature on decision support systems (DSS) for fleet scheduling is limited. There exists a plethora of research concerning optimization for specific companies on a case-by-case basis, but it is difficult to find many descriptions of long term, flexible solutions. Consequently, there exists a need for research outlining a possible approach to evaluate the effectiveness of maritime DSS software. My work contributes by providing such approach.

Additionally, my work addresses specific business needs of my partner company who can use the outcomes of the research to understand the effects of their product, as well as utilize those insights in the areas of developing a marketing and pricing strategy. However, those business needs can be relevant for the whole maritime software sector. Providing insights to contribute to understanding the risk in switching to a more advanced, digital solution can be beneficial both for maritime software companies, as well as maritime transportation companies.

6.4 Limitations and further research

One limitation of the study is that no learning curve in software use is assumed. For this thesis there was no available data on how quickly the company switched to the new system and whether they still relied on old methods and processes. The learning effect curve could be estimated for example by interviewing a sufficiently large number of schedulers to determine how quickly they adapted to the use of new software. This was outside of the scope of this thesis, but such additional interview-based information could be used to divide the ex-post period into time-wise subdivisions indicating different levels of software use. Those levels of software use could be used as weights in an ensemble model similar to the models applied and developed in this thesis.

Another limitation of this study is that it uses historical data from only one customer who purchased and applied only one specific software solution. As more maritime transportation companies shift towards digital and data-driven solutions and generate historical maritime pre-event and post-event data, future research could use this new information to widen the scope of the analysis. This thesis provides a framework to analyze changes in maritime KPIs in the context of a shipping company switching to a new software, which is a major change in the way of doing business. Researching impact of a specific software on a larger number of maritime shipping companies or researching the impact of different examples of maritime scheduling software could provide further contributions in the field of maritime data analytics.

In addition, it is important to point out that the software which effect was studied is continuously being developed. Therefore, its impact can be different as time progresses. It would be interesting to perform this kind of event impact analysis several times by collecting data for periods after new major milestones are implemented, for example after the implementation of mathematical optimization algorithms. The achieved results could be then compared with the results of this study, which assessed the impact of maritime software which over the studied period prioritized UX rather than optimization-based recommendation mechanism. This idea of periodic analysis is closely associated with the findings of Fagerholt (2004) described in more detail in Section 3.1. Fagerholt recommends that maritime DSS software should be implemented in stages. An event impact analysis, such as the one proposed in this thesis, could be performed at each of the major development stages. This way, it would be possible to assess the impact of a specific major improvement in the software, and insights from such assessment could play a role in the software company's strategy to market this improvement. Moreover, such periodic analysis could also be used to influence the direction of future development of the software DSS product.

Additionally, an assumption is made that this research can be relevant for the maritime software and maritime transportation industries because it provides numerical analysis results which can be used to drive business value. However, it is not obvious that providing such data translates to a higher business value for marketing or pricing applications better than qualitative, descriptive, non-data-driven argumentation. This could be an area of research on the crossroads of business and psychology.

Lastly, it would be interesting to further experiment with different validation approaches. A validation set approach is the simplest and the most computationally efficient. For this reason, it was the go-to approach given the limited resources. A more complex validation method, for example K-fold validation could further minimize validation set RMSE and make the model more precise. However, the deployed RF models showed that the effect of incremental improvements in validation set RMSE scores corresponded to incrementally smaller changes in the predicted KPI. Due to the observed damping behavior of the incremental RMSE improvements it is not obvious that a model with a better RMSE score would achieve significantly better predictions. There is a trade-off between prediction accuracy and model computational requirements, and the outcome of this analysis which showed converging behavior in estimated software effect suggests that further RMSE improvements would only lead to almost the same final results.

7 Conclusion

Maritime shipping is changing towards becoming a more digitalized, data driven industry. Recent application of scheduling and fleet allocation decision support systems, which appear simpler and quicker than manual practices, raises the question whether those software solutions can achieve as good of a result as more complicated traditional methods involving spreadsheets and manual calculations.

I apply methods from Wang et al. (2019) in a new setting to solve a timely business problem of determining the effect of implementing a scheduling and fleet allocation DSS on vessel tonnage utilization. Subsequently, I build on top of that framework and extend it to create original methods of approaching the problem. My method provides an in-depth analysis of the specific business challenge; it can also be generalized to contribute to the general body of knowledge in the area of maritime technology research and event impact analysis.

The results of this research indicate that implementing the maritime DSS did not create any substantial operational risk for the customer company. On the other hand, it also did not visibly improve their performance as measured by the tracked KPI. However, because in the studied period fleet optimization algorithms were not a part of the DSS, it is difficult to expect dramatic changes which could be expected in the case of an extensive use of mathematical optimization. Instead, in connection with the time saving aspect of the DSS as well as the benefits of standardization, scalability, and increased ease of future implementation of mathematical optimization, a software not having a negative effect on the industry specific KPIs can be considered successful. Incorporating FAS software in the everyday operations of the company can be seen as groundwork, a base of building organizational knowledge and understanding of technology. This base can be expected to provide a smoother transition to increasingly optimization-based, data-driven fleet scheduling and allocation methods, essentially “future-proofing” the shipping company.

References

Adams, R. P. and MacKay, D. J. (2007). Bayesian online changepoint detection. *arXiv preprint arXiv:0710.3742*.

Arnholt, A. T. and Evans, B. (2021). *BSDA: Basic Statistics and Data Analysis*. R package version 1.2.1.

Belgiu, M. and Drăguţ, L. (2016). Random forest in remote sensing: A review of applications and future directions. *ISPRS journal of photogrammetry and remote sensing*, 114:24–31.

Bonet, I., Peña, A., Lochmuller, C., Patiño, H. A., Chiclana, F., and Góngora, M. (2021). Applying fuzzy scenarios for the measurement of operational risk. *Applied Soft Computing*, 112:107785.

Box, G. E. and Tiao, G. C. (1975). Intervention analysis with applications to economic and environmental problems. *Journal of the American Statistical association*, 70(349):70–79.

Dataloy Systems AS (2022). *Fleet Plan Allocation and Scheduling*. Available at https://dataloy-systems.com/dataloy-fas/.

Digital Ship (2021). Q88 and dataloy systems partner on data-driven solution. Available at https://thedigitalship.com/news/maritime-software/item/7099-q88-and-dataloy-systems-partner-on-data-driven-solution.

Diz, G., Scavarda, L. F., Rocha, R., and Hamacher, S. (2014). Decision support system for petrobras ship scheduling. *Interfaces*, 44(6):555–566.

Duru, O., Bulut, E., Huang, S., and Yoshida, S. (2013). Shipping performance assessment and the role of key performance indicators (kpis):'quality function deployment'for transforming shipowner's expectation. *Available at SSRN 2195984*.

El Machkouri, M. (2011). Asymptotic normality of the parzen–rosenblatt density estimator for strongly mixing random fields. *Statistical Inference for Stochastic Processes*, 14(1):73–84.

Fagerholt, K. (2004). A computer-based decision support system for vessel fleet scheduling—experience and future research. *Decision Support Systems*, 37(1):35–47.

Fagerholt, K. and Lindstad, H. (2007). Turborouter: An interactive optimisation-based decision support system for ship routing and scheduling. *Maritime Economics & Logistics*, 9(3):214–233.

Greenwell, B. M. and Boehmke, B. C. (2020). Variable importance plots—an introduction to the vip package. *The R Journal*, 12(1):343–366.

Grolemund, G. and Wickham, H. (2011). Dates and times made easy with lubridate. *Journal of Statistical Software*, 40(3):1–25.

Jarušková, D. (1997). Some problems with application of change-point detection methods to environmental data. *Environmetrics: The official journal of the International Environmetrics Society*, 8(5):469–483.

Koski, A., Siren, R., Vuori, E., and Poikolainen, K. (2007). Alcohol tax cuts and increase in alcohol-positive sudden deaths—a time-series intervention analysis. *Addiction*, 102(3):362–368.

Kuhn, M. and Wickham, H. (2020). *Tidymodels: a collection of packages for modeling and machine learning using tidyverse principles.*

Lagarde, M. (2012). How to do (or not to do)… assessing the impact of a policy change with routine longitudinal data. *Health policy and planning*, 27(1):76–83.

Learn UI Design (2022). *Data Color Picker*. Available at https://learnui.design/tools/data-color-picker.html#palette.

Lehmann, E. L., Romano, J. P., and Casella, G. (2005). *Testing statistical hypotheses*, volume 3. Springer.

Lüdecke, D. (2018). sjmisc: Data and variable transformation functions. *Journal of Open Source Software*, 3(26):754.

Marintek (2008). *Publications within maritime logistics and the TurboRouter development.* Available at https://www.sintef.no/projectweb/turborouter/publications/.

Mbugua, J. K., Bloom, G. H., and Segall, M. M. (1995). Impact of user charges on vulnerable groups: the case of kibwezi in rural kenya. *Social science & medicine*, 41(6):829–835.

Moosa, I. A. (2007). Operational risk: a survey. *Financial Markets, Institutions & Instruments*, 16(4):167–200.

Parzen, E. (1962). On estimation of a probability density function and mode. *The annals of mathematical statistics*, 33(3):1065–1076.

R Core Team (2021). *R: A Language and Environment for Statistical Computing.* R Foundation for Statistical Computing, Vienna, Austria.

Rosenblatt, M. (1956). Remarks on Some Nonparametric Estimates of a Density Function. *The Annals of Mathematical Statistics*, 27(3):832 – 837.

Saunders, L. J., Russell, R. A., and Crabb, D. P. (2012). The coefficient of determination: what determines a useful r2 statistic? *Investigative ophthalmology & visual science*, 53(11):6830–6832.

Sievert, C. (2020). *Interactive Web-Based Data Visualization with R, plotly, and shiny.* Chapman and Hall/CRC.

Smirnov, N. V. (1939). On the estimation of the discrepancy between empirical curves of distribution for two independent samples. *Bull. Math. Univ. Moscou*, 2(2):3–14.

Trosset, M. W. (2009). *An introduction to statistical inference and its applications with R.* Chapman and Hall/CRC.

Veson Nautical (2022a). *Analytics.* Available at https://veson.com/veson-imos-platform/analytics/.

Veson Nautical (2022b). *Western Bulk Chooses Veson Nautical after Data-Driven Review.* Available at https://veson.com/success_story/western-bulk/.

Wallot, S. and Leonardi, G. (2018). Deriving inferential statistics from recurrence plots: A recurrence-based test of differences between sample distributions and its comparison to the two-sample kolmogorov-smirnov test. *Chaos: An interdisciplinary journal of nonlinear science*, 28(8):085712.

Wang, Q., Farahat, A., Ristovski, K., Gupta, C., and Zheng, S. (2019). Evaluation of event impact on key performance indicators. In *2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA)*, pages 726–733. IEEE.

Wickham, H., Averick, M., Bryan, J., Chang, W., McGowan, L. D., François, R., Grolemund, G., Hayes, A., Henry, L., Hester, J., Kuhn, M., Pedersen, T. L., Miller, E., Bache, S. M., Müller, K., Ooms, J., Robinson, D., Seidel, D. P., Spinu, V., Takahashi, K., Vaughan, D., Wilke, C., Woo, K., and Yutani, H. (2019). Welcome to the tidyverse. *Journal of Open Source Software*, 4(43):1686.

Zhou, X., Zhu, X., Dong, Z., Guo, W., et al. (2016). Estimation of biomass in wheat using random forest regression algorithm and remote sensing data. *The Crop Journal*, 4(3):212–219.