improving the NS train schedule by working on train turnaround times

Utrecht University

Master of Science in Applied Data Science

First Supervisor: J.A. Hoogeveen

Second Supervisor: Feelders, Ad

Host Institute: NS (Nederlandse spoorwegen)

Parisa Safi

Student number: 2985160

July 13st 2023

Table of Content

1. Introduction

This chapter revolves around the reason why this report was conducted. It describes the problem, and goals of this paper.

This study was carried out as an assignment for the NS company.

Nederlandse Spoorwegen (NS; English: "Dutch Railways") is the principal passenger railway operator in the Netherlands. It is a Dutch state-owned company founded in 1938. The Dutch rail network is the busiest in the European Union, and the third busiest in the world after Switzerland and Japan.

NS runs 4,800 scheduled domestic trains a day, serving 1.1 million passengers. The NS also provides international rail services from the Netherlands to other European destinations and carries out concessions on some foreign rail markets through its subsidiary Abellio.

The NS covers most of the country, with almost all cities connected, mostly with a service frequency of two trains an hour or more and at least four trains per hour between all of the largest five cities (Amsterdam, Rotterdam, The Hague, Utrecht and Eindhoven) as well as some smaller cities (Nijmegen, Amersfoort, Arnhem, 's-Hertogenbosch, Dordrecht and Leiden) [1].

The train timetable serves as the fundamental framework for organizing train operations and concurrently serves as an efficient means to present a comprehensive outline of transportation production activities. It is extensively acknowledged that the effectiveness of the train timetable directly influences the operational safety, passenger contentment, and economic advantages of the rail transit system. The evaluation of train timetable performance holds immense significance as it has the potential to furnish valuable and enlightening insights on enhancing service quality. [2] In light of the swift expansion of rail transit systems, there has been a growing scholarly focus on the field of train scheduling. Scholars have dedicated their efforts to optimizing train schedules with the aim of assisting dispatchers in making informed decisions that align with the expectations of both passengers and operators. The primary objectives of such optimization endeavors primarily revolve around augmenting passengers' satisfaction levels, specifically by reducing overcrowding, minimizing passengers' waiting times and optimizing passengers' travel durations. [3] Conversely, these objectives also encompass the reduction of operational costs,

which include energy consumption , train-related expenses , and train travel times , among other pertinent factors. [4]

To find a solution for robust timetable, this thesis examines the possibilities of using machine learning to predict and optimize the turnover time. By doing so, it offers a chance to preparation for the designers in train companies to implement a set of rules for timetable in easier and effective way.

Turnover time prediction optimization of train events is important for proactive decision-making for both operators and passengers.

For example, accurate train arrivals/departures prediction allows operators to accurately estimate train operation status and make informed decisions when planning a timetable or rescheduling trains in the event of service disruptions.

In this project We know the minimum technical turnover time, For robustness purposes, we want to plan more than this minimum technical turnover time.

1.1 Goals:

The objective of this study is to facilitate the planning of a resilient train schedule to assist train travelers. To achieve this goal, we recognize the significance of acquiring a deeper understanding of the time required for a train to complete a turnaround process. A turnaround refers to the departure of a train from a station in the same direction as its arrival. In order to ensure robustness in train scheduling, it is imperative that the turnaround time be accomplished without delay in at least 90% of cases. In pursuit of enhanced robustness, we aim to devise a schedule that incorporates a turnover time exceeding the minimum technical requirement.

Main question:

How much turnover time should we plan to gain a robust timetable?

Aimed product:

easy to implement set of rules for timetable designers

2. Literature Reviews

Within the domain of train operation prediction, the majority of research endeavors

have concentrated their attention on the prognostication of train delays through the utilization of machine learning methodologies. As an illustrative instance, Li et al. [5] formulated a random forest regression model with the intention of forecasting imminent train delays within the network of the Dutch railways.

Markovic et al. [6] conducted a study aimed at examining the effects of various infrastructure projects on train delays. In order to ascertain the relationship between passenger train arrival delays and different railway infrastructures, the researchers employed support vector regression and analyzed data obtained from Serbian Railways.

predict train delays for passenger rails in Deutsche Bahn, Nair et al.[7] This model integrated multiple components, including a random forest algorithm to consider network traffic conditions, kernel regression to incorporate train-specific dynamics, and a mesoscopic simulation model to incorporate variations in travel and dwell times. The ensemble model was devised as a comprehensive approach to comprehensively capture the multifaceted factors influencing train delays .

Li et al. [8] introduced a novel approach employing Extreme Learning Machines (ELM) that were fine-tuned using particle swarm optimization (PSO) for the prediction of train arrival delays on the Beijing-Kowloon High-Speed Rail (HSR) line in China. The utilization of the PSO algorithm alleviates the laborious task of manually adjusting hyperparameters associated with the ELM model, thus enhancing the efficiency and effectiveness of the prediction process. Oneto et al. [9] presented a dynamic system for predicting train delays, which leverages both Shallow and Deep Extreme Learning Machines (ELM) in conjunction with state-of-the-art in-memory large-scale data processing technologies. The proposed system offers an innovative approach that integrates both shallow and deep learning models, along with advanced data processing capabilities, to enhance the accuracy and timeliness of train delay predictions.

The empirical investigation conducted on the Italian railways demonstrated that the incorporation of weather data resulted in improved accuracy. In their research, Wen et al. [10] employed Long Short-Term Memory (LSTM) to forecast train arrival delays, considering the influence of train interactions and the propagation of delays. The findings of this study indicated that the LSTM model exhibited superior performance compared to both the random forest model and the artificial neural network model.

Huang et al. [11] introduced the CLF-Net model for predicting train delays, which

incorporates three components: 3-dimensional convolutional neural networks (3D CNN), fully connected neural network (FCNN), and Long Short-Term Memory (LSTM). This model was devised to address the intricate nature of the data generated by moving trains. Spatial-temporal features were inputted into the 3D CNN, time-series variables were fed into the LSTM, and non-time series factors were simultaneously processed by the FCNN. By segregating the processing of different features, only minimal errors were detected in both the overall railway line and at individual stations, as evidenced by the findings of this study.

Several studies predict other train events such as dwell times at stations Li, Daamen, and Goverde [12] ; running times between stations Gorman [13] , Huang et al. [14] ; arrival times at stations [15] using machine learning approaches.

3. Data

3.1 Data collection:

The data consolidation process involved merging two CSV files. The first file, referred to as the "Basis Execution Data," encompassed the data collected by the NS railway company in the Netherlands from October to December 2022, encompassing all stations across the country. The second file corresponded to a similar database, with the exception that it included information regarding driver changes.

3.2. Data preparation:

Prior to the commencement of the training phase, the data from the two sources were consolidated to facilitate its utilization as input for machine learning purposes. For each unique train number and date combination, the pertinent details regarding the departure and arrival times were extracted. Subsequently, the turnover time was calculated, and the relevant features pertaining to the turnover time were determined. These pieces of information were amalgamated and organized into a novel dataset named "merged_data," where each entry contained information about a departure as well as its corresponding turnover time. These datasets were used to create a data frame that would be useful for different analyses .

Departures lacking train number data were excluded from the dataset, and departures with an OMNUMMERING number exceeding 2700 were also disregarded. This decision was motivated by the necessity to eliminate outlier data, thus enabling the

prediction of turnover time and subsequent optimization.

This paragraph will discuss the different datasets and the features in it.

Basis Execution Data:

The dataset in question comprised numerous columns predominantly pertaining to train attributes. Consequently, a judicious choice was made to select specific columns that provided essential information, encompassing details such as departure and arrival times, station information, material numbers, train numbers, dates, peak or off-peak periods, train length, and the type of rolling stock. Furthermore, any missing values within the material numbers were eliminated. This step was undertaken with the objective of mitigating data errors and reducing the computational burden during the process of merging datasets.

Set 3 + Driver Change Included:

In this dataset, along with the existing train-related features, the inclusion of the driver change column provided further insight into the dynamics of train operations. This additional information allows for a more comprehensive analysis of factors influencing turnover time and provides a basis for exploring the impact of driver changes on overall system performance.

Table 1. Variables in the merged dataset

Variable name	Type
SL_RIJRICHTINGKERING_IND	int64
SL_OMNUMMERING	float64
SL_DRGLPT	object
SL_TREINNR	int64
SL_VERKEERSDATUM	object
SL_BASIC_UITVTIJD_VERTREK	datetime64[ns]
SL_MAT_LENGTE	float64
Wisselmachinist	float64
verkeersdatum	object
bewegingcode	float64
act_drglptVolgend	object
act_actSoortVolgend	object

act_uitvoertijdVolgend	datetime64[ns]
spitstype	object
matSoort	object
aantal_bakken	float64
week	float64
act_OplantijdVolgend	datetime64[ns]
turn over time	timedelta64[ns]

This final data frame contained 156710 rows and 22 columns.

3.3. feature selection:

Feature selection encompasses the process of identifying significant input variables, aiming to eliminate redundant or irrelevant features. Many scholarly works have relied on subjective approaches, leveraging domain knowledge and common sense, to determine the selection or development of variables to be incorporated into models deemed to possess predictive value.[16]

This process serves to mitigate computational complexity and enhance model performance. In this phase, Exploratory Data Analysis (EDA) is employed to thoroughly examine the dataset, aiming to uncover underlying patterns and facilitate the formulation of logical assumptions and hypotheses regarding the impact of specific factors on the target variable in alignment with the research question.

3.4. Exploratory Data Analysis:

After performing data cleaning on the dataset, a careful evaluation of the available features was conducted to determine their potential usefulness. Subsequently, visualizations were employed to gain insights into the relationships between these features and the turnover time. The purpose of this exploratory data analysis was to select the most relevant features from the dataset for implementation in the subsequent steps of the methodology. It is worth noting that all visualizations were generated based on the merged_data dataset.

Table 2 presents a comprehensive overview of the prediction variables utilized in the study, along with their respective data sources. The primary data source for developing the turnover time prediction model is the train operation data. This dataset encompasses various factors associated with train operations, such as crew scheduling (including driver changes), the type of rolling stock, peak and off-peak periods, train length, and the number of wagons. These variables were selected based on their perceived relevance in influencing turnover time.

Table 2. selected Variables

Variable_name	**Type**
SL_MAT_LENGTE	float64
Wisselmachinist	float64
spitstype	object
matSoort	object
aantal_bakken	float64

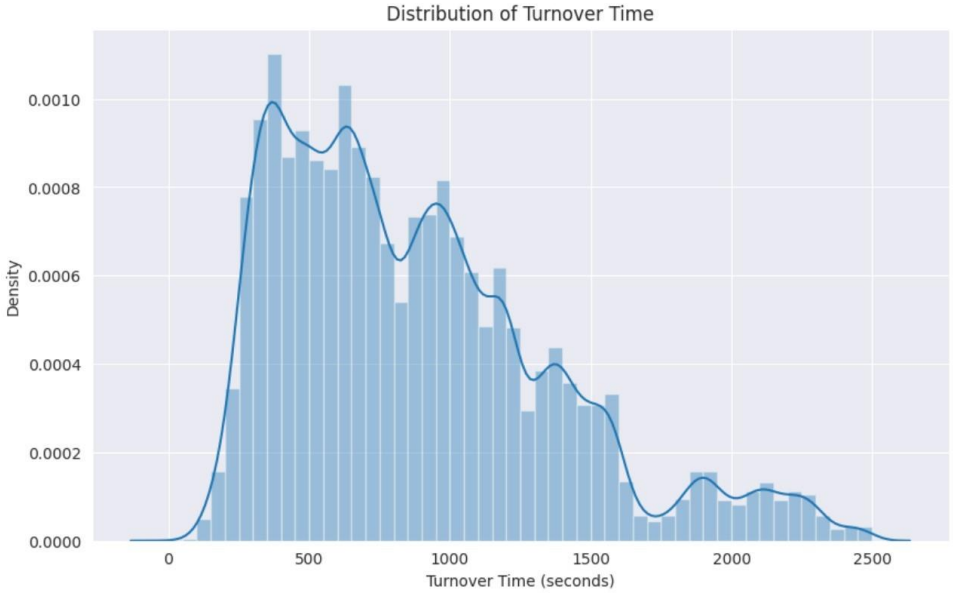
3.5. visualization:

During the exploratory analysis, correlations were found between certain variables.

Firstly, Figure 1 illustrates that the majority of turnover time values fall within the range of 400 seconds to nearly 1000 seconds. This observation implies that the typical duration for trains to change direction at stations spans approximately 5 to 11 minutes.

(The actual range and interpretation of the turnover time values may vary based on the specific context and dataset being analyzed.)

Figure 1. Distribution of turnover time



Secondly, in figure 2, the most frequent occurrences of train delays are observed within the turnover time category of 10 to 20 minutes. This finding indicates that trains with turnover time values falling between 10 to 20 minutes experience delays more frequently.

Delays are determined by calculating the time difference between a train's arrival at the station and its departure. If this time difference exceeds 59 seconds, it is considered as a delay.

Figure 2

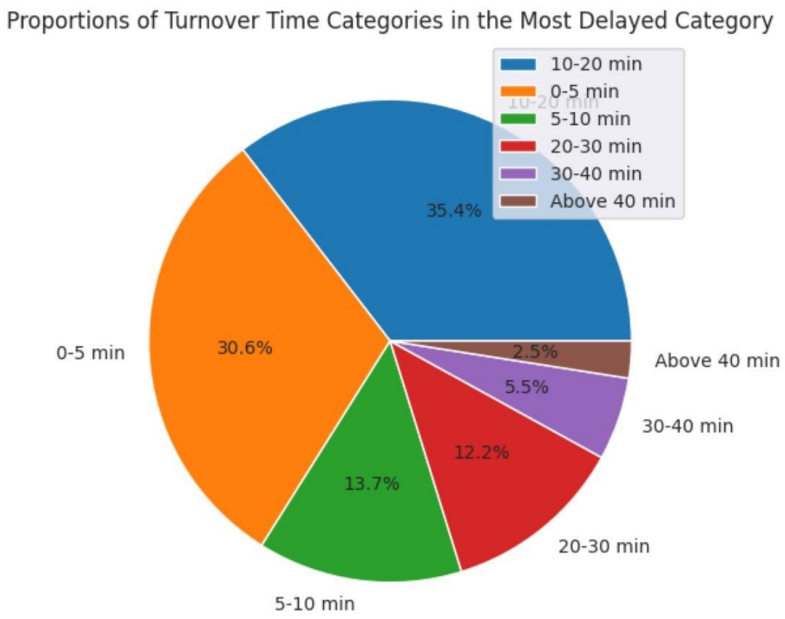
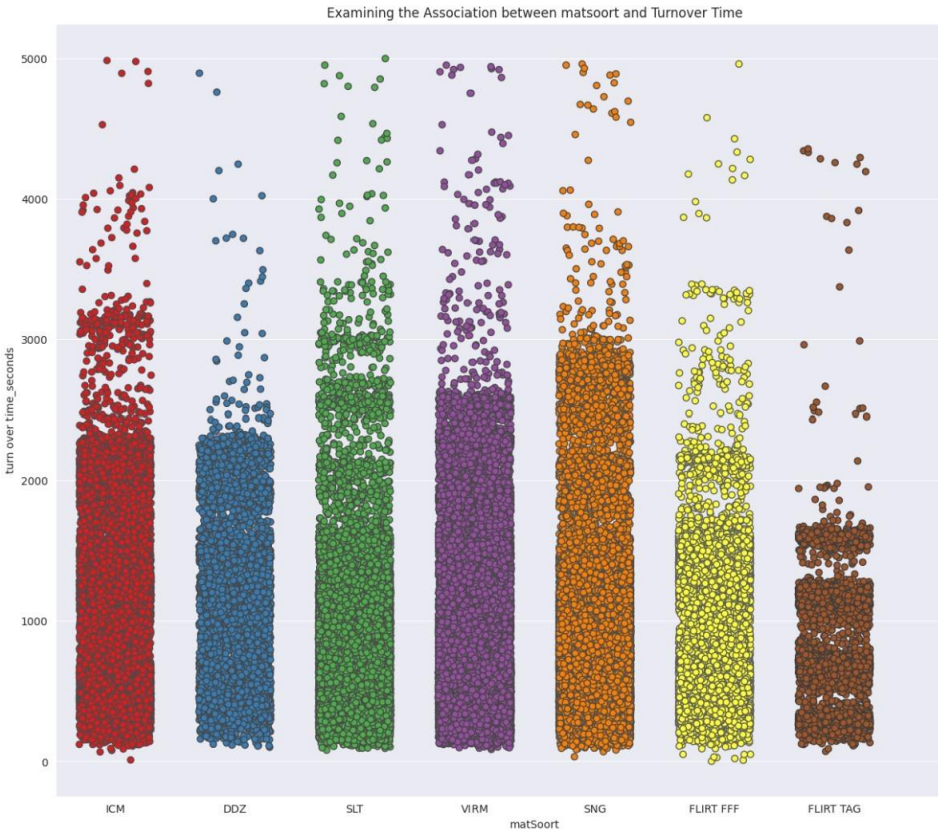


Figure 3 presents a strip plot the relationship between turnover time and the type of rolling stock. The stripplot facilitates the identification of patterns, outliers, and the overall distribution of turnover time within each category of rolling stock. Each mark in the strip plot represents a data point and is positioned along the axis based on its corresponding turnover time value. Horizontal spreading of the marks is implemented to prevent overlapping, ensuring a clear visualization of the distribution.

Figure 3



These figures 4, 5 examine the association between train length and turnover time in relation to driver change patterns. The plot illustrates the relationship between turnover time, train length, and the occurrence of driver change.

In the case where the driver remains unchanged, a positive correlation is observed between turnover time and train length, indicating an upward slope. However, when considering the influence of the driver change variable, a contrasting trend is observed. Specifically, the relationship between turnover time, train length, and driver change exhibits a negative correlation, denoted by a downward slope.

Figure4

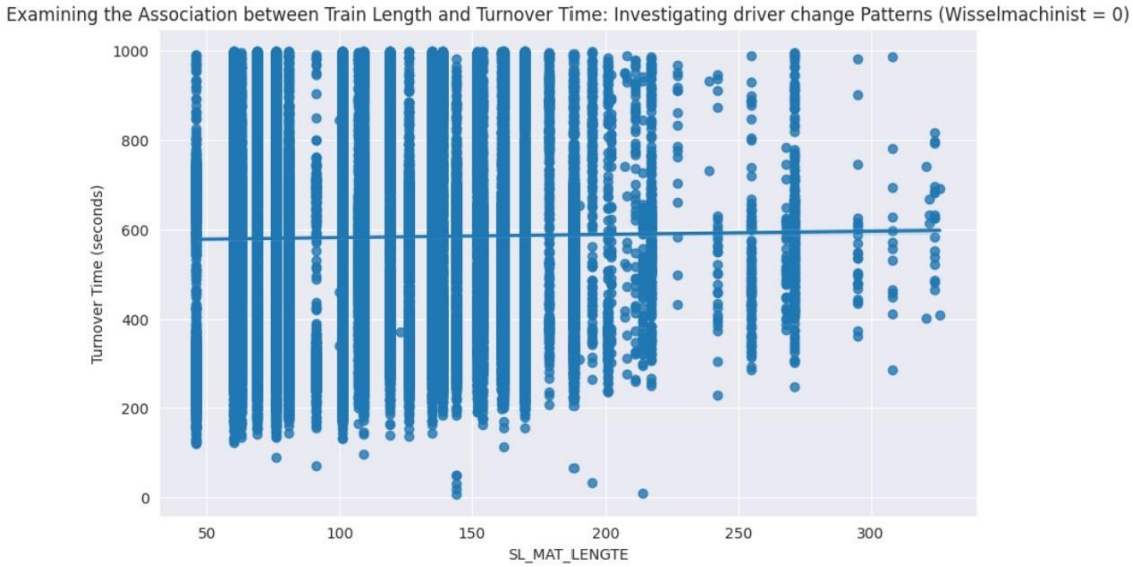
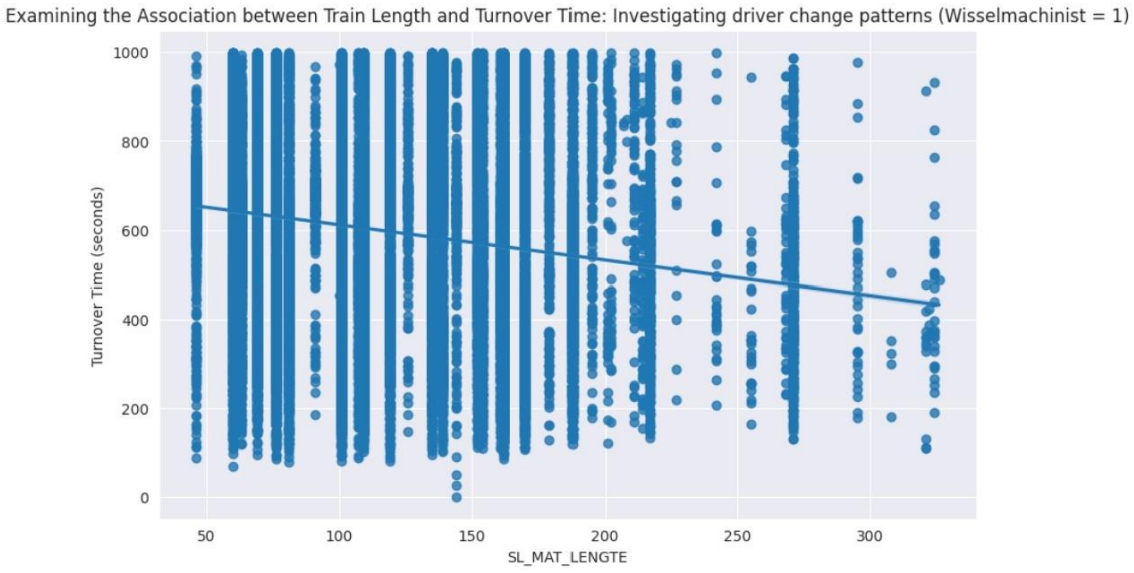


Figure5



4. **Methods and Evaluation:**

Linear regression analyses are predominantly employed to ascertain the influence of variables on train delays, as well as to make real-time predictions regarding them. Utilizing linear regression, one can forecast the duration of a train's journey while

identifying the principal factors associated with congestion that exert the greatest impact on delay. This regression model is straightforward to implement, yielding easily interpretable results. Nevertheless, it possesses limitations when it comes to modeling the intricate non-linear relationships between input and output variables.

One drawback inherent in statistical regression is its tendency to disregard the intricate relationships existing among various variables.

Machine learning (ML) is utilized to uncover latent knowledge by leveraging the relationships present within historical data, thus facilitating the generation of reliable and replicable predictions.

Definitive evidence regarding the superiority of specific ML algorithms over others is yet to be established, and the determination of the optimal model for predicting turnover time ultimately relies on the comparison of different algorithms.[16]

Linear regression is a supervised learning method that is widely used for predicting the outcomes or analysing the association between variables. The first assumption of this method is that there exists a linear relationship (formula 1) between the predictor variables (xi) and the response variable (Y).

(formula 1)

Multiple linear regression , $Y = \beta 0 + \beta 1 \quad x1 + \beta 2x2 + \cdot \cdot \cdot + \beta pxp + \epsilon$

Where βi are unknown constants, representing the model coefficient, and ϵ is a mean-zero random error term. We typically assume that the error term is independent of the predictor variable.

Furthermore, in order to assess the results of the estimated coefficients, t-test can be performed on the coefficients. Choosing a confidence interval of 95%, a p-value smaller than 5% indicates that such a substantial association between the predictor and the response is unlikely to be observed due to chance, and there is a statistically meaningful association between the predictor and the response. [17]

To evaluate the effectiveness of the model, a random division was performed, with 2/3 of the data allocated for training and 1/3 for testing . To assess the model's performance, R-squared value was calculated.

The results showed a R-squared of 0.02. Considering that the variables were z-normalized, This represents the coefficient of determination, which measures the proportion of variance in the dependent variable (turn over time_seconds) that can be explained by the independent variables in the model. In this case, the R-squared value is 0.025, indicating that approximately 2.5% of the variance in the dependent variable is explained by the independent variables. This low R2 value suggests that the predictor variables may not be effectively capturing the underlying factors that influence the response variable. Therefore, the model may not be a good fit for the data, and alternative approaches may need to be considered to improve the model's performance and increase the proportion of variance explained.

These are the estimated coefficients for each independent variable. They represent the expected change in the dependent variable associated with a one-unit change in the corresponding independent variable, while holding other variables constant. The coefficients indicate the direction and magnitude of the relationship.

SL_MAT_LENGTE: The coefficient is 0.0412 It indicates that for a one-unit increase in the variable "SL_MAT_LENGTE," the predicted value of the target variable (response) increases by approximately 0.0412 units. This coefficient has a standard error of 0.005, suggesting the precision of the estimate. The t-value (7.988) and associated p-value (0.000) indicate that this coefficient is statistically significant, meaning it is unlikely to have occurred by chance.

Wisselmachinist: The coefficient is 0.0937. It suggests that for a one-unit increase in the variable "Wisselmachinist," the predicted value of the target variable increases by approximately 0.0937 units. The standard error is 0.003, and the high t-value (29.878) and very low p-value (0.000) indicate that this coefficient is highly statistically significant.

matSoort: The coefficient is –0.0773. It signifies that for a one-unit increase in the variable "matSoort," the predicted value of the target variable decrease by approximately –0.0773 units. The standard error is 0.003, and the t-value (-23.033) and p-value (0.000) indicate that this coefficient is statistically significant.

aantal_bakken: The coefficient is –0.0835. It implies that for a one-unit increase in the variable "aantal_bakken," the predicted value of the target variable increases by approximately 0.0835 units. The standard error is 0.005, and the t-value (-23.033) and p-value (0.000) suggest that this coefficient is statistically significant.

spitstype: The coefficient is –0.0845. It indicates that for a one-unit increase in the variable "spitstype," the predicted value of the target variable decreases by approximately 0.0845 units. The negative sign implies a negative relationship. The standard error is 0.003, and the t-value (-26.625) and p-value (0.000) indicate that this coefficient is statistically significant.

Overall, these coefficients represent the estimated effects of each predictor variable on the turnover time in the multiple linear regression model. They help understand the direction and magnitude of the relationship between the predictors and the target variable, while the standard errors, t-values, and p-values provide information about the statistical significance of each coefficient.

	coef	std err	t	P>|t|	[0.025	0.975]
SL_MAT_LENGTE	0.0412	0.005	7.988	0.000	0.031	0.051
Wisselmachinist	0.0937	0.003	29.878	0.000	0.088	0.100
matSoort	-0.0773	0.003	-23.033	0.000	-0.084	-0.071
aantal_bakken	-0.0835	0.005	-16.541	0.000	-0.093	-0.074
spitstype	-0.0845	0.003	-26.625	0.000	-0.091	-0.078

Omnibus:	185392.150	Durbin-Watson:	2.011
Prob(Omnibus):	0.000	Jarque-Bera (JB):	1582360403.375
Skew:	11.981	Prob(JB):	0.00
Kurtosis:	605.474	Cond. No.	3.11

Figure6: the evaluation of model

I am intrigued by a particular observation regarding the discrepancy between the performance of a model on the training data and the subsequent evaluation on the test data. Specifically, when examining the R-squared metric, it was noted that the R-squared value achieved on the training set was 0.61, indicating a moderately strong fit between the model and the training data.

When the R-squared value of the train set is significantly better than the R-squared value of the test set, it typically indicates a situation of overfitting. Overfitting occurs when a model performs extremely well on the training data but fails to generalize well to new, unseen data.

Here's what might be happening:

Overfitting: The model has learned the patterns and noise in the training data too well, to the point that it starts capturing random fluctuations or noise specific to the training set. As a result, it doesn't generalize well to new data, leading to a lower R-squared value on

the test set. The model becomes too specific to the training set and fails to capture the underlying relationships in the broader population.

Lack of generalization: The model might be too complex or have too many parameters relative to the amount of training data available. This excessive complexity allows the model to fit the training data closely but makes it difficult to generalize to new data. The model may be "memorizing" the training set instead of learning the underlying patterns.

Data mismatch: There might be differences in the characteristics or distribution of the training and test sets. If the test set differs significantly from the training set in terms of data distribution, relationships, or other relevant factors, the model may struggle to perform well on the test set, resulting in a lower R-squared value.

To address this issue, I apply Regularization techniques, Consider using LASSO methods to add a penalty term that discourages excessive complexity and helps prevent overfitting.

after I apply LASSO model I determined the best alpha and it get me 0.0001 that "alpha" refers to the regularization parameter, also known as the penalty term. It controls the amount of regularization applied to the model. Specifically, a smaller value of alpha imposes a weaker penalty, allowing the model to fit the training data more closely.

as mentioned above, that the best alpha in Lasso is 0.0001, it means that there is a relatively small value for the regularization parameter that yielded the optimal performance for the model. This value suggests that I am applying a relatively mild regularization, allowing the model to capture more fine-grained details from the training data. However, it's important to note that the optimal value of alpha may vary depending on the specific dataset and the complexity of the problem at hand.

The LASSO regularization technique is commonly employed to facilitate feature selection by driving the coefficients associated with less relevant or redundant variables towards zero. LASSO encourages sparsity in the coefficient estimates, effectively resulting in the exclusion of certain predictors by setting their coefficients to exactly zero. This attribute of LASSO is particularly advantageous in scenarios where the objective is to identify the most influential predictors and streamline the model.

By eliminating specific coefficients, LASSO aids in the identification of a subset of predictors that exert a considerable impact on the target variable, while disregarding those that contribute less significantly. This not only enhances the interpretability of the model but also mitigates the risk of overfitting by reducing model complexity and

restraining its reliance on extraneous predictors or noise.

However, in the current study, when applying LASSO, none of the coefficients were found to approach zero. This suggests the presence of strong signals from all predictors in the model. The absence of coefficients being shrunk to zero indicates that each predictor substantially contributes to explaining the variations observed in the target variable, rendering it challenging to differentiate between important and unimportant predictors.

5.conclusion:

The implementation of a multiple linear regression model was found to be inadequate in optimizing turnover time and addressing delays and enhancing the robustness of the timetable.

linear regression does not yield satisfactory accuracy, and applying LASSO regularization fails to improve the results, it suggests several possibilities:

Linear regression is predicated on the assumption of a linear relationship between predictor variables and the target variable. However, if the actual relationship is non-linear in nature, employing LASSO regularization alone may prove insufficient in resolving this underlying concern. In such instances, alternative regression models capable of accommodating non-linear relationships, such as polynomial regression or non-linear regression, may present more suitable alternatives. These models possess the capacity to capture and characterize the intricate non-linear associations that exist within the data, thus providing more accurate and comprehensive modeling outcomes.

Data limitations pose a substantial impact on the efficacy of models, including linear regression with LASSO. The quality and representativeness of the dataset play crucial roles in determining model performance. Inadequacies such as the absence of pertinent predictors, the presence of noisy or unreliable data, or the imposition of limitations in sample size can impede the model's capacity to achieve desirable levels of accuracy. Thus, the interpretability and generalization capability of the model are contingent upon the availability of a high-quality dataset that encompasses essential variables and minimizes the influence of extraneous factors.

Moreover, the underlying complexities within the data may surpass the capabilities of a

simplistic linear model, even when coupled with regularization techniques. Complex relationships or interactions among predictors may exist, which defy adequate representation through a linear model. In such instances, alternative and more sophisticated modeling techniques, including decision trees, random forests, or neural networks, warrant consideration. These approaches offer greater flexibility and the ability to capture intricate patterns, enabling a more accurate representation of the underlying relationships within the data.

6. Future Works:

In future research endeavors, it is imperative to conduct data collection with heightened attention to detail and inclusivity of a broader array of potential variables. This approach will contribute to a more comprehensive and nuanced understanding of the research domain, allowing for enhanced analysis and interpretation of the collected data. By expanding the scope of data collection to encompass additional variables, researchers can capture a more comprehensive representation of the underlying factors influencing the phenomenon under investigation. Such an approach facilitates a more robust examination of the relationships, dependencies, and complexities within the dataset, thereby enriching the research outcomes and contributing to a more holistic understanding of the subject matter.

7. Bibliography

[1] Wikipedia contributors, “Nederlandse spoorwegen,” *Wikipedia*, Jun. 2023, [Online].

Available: https://en.wikipedia.org/wiki/Nederlandse_Spoorwegen#cite_note-4

[2] F. Liu, R.-H. Xu, W. Fan, and Z. Jiang, “Data analytics approach for train timetable performance measures using automatic train supervision data,” *Iet Intelligent Transport Systems*, vol. 12, no. 7, pp. 568–577, Mar. 2018, doi: 10.1049/iet-its.2017.0287.

[3] X. Tian, An optimization of train scheduling for urban rail transits under time-dependent conditions [Ph.D. thesis], Lanzhou Jiaotong University, 2013.

[4] X. Zhou and M. Zhong, “Single-track train timetabling with guaranteed optimality: Branch-and-bound algorithms with enhanced lower bounds,” *Transportation Research Part B-methodological*, vol. 41, no. 3, pp. 320–341, Mar. 2007, doi: 10.1016/j.trb.2006.05.003.

[5] ZhongCan Li et al. “Near-term train delay prediction in the Dutch railways network”. In: International Journal of Rail Transportation 9.6 (2021), pp. 520–539.)

[6] Nikola Markovic et al. “Analyzing passenger train ´ arrival delays with support vector regression”. In: Transportation Research Part C: Emerging Technologies 56 (2015), pp. 251–262.

[7] R. R. Nair *et al.*, “An ensemble prediction model for train delays,” *Transportation Research Part C-emerging Technologies*, vol. 104, pp. 196–209, Jul. 2019, doi: 10.1016/j.trc.2019.04.026.

[8] Y. Li, X. Xu, J. Li, and R. Shi, *A delay prediction model for high-speed railway: an extreme learning machine tuned via particle swarm optimization*. 2020. doi: 10.1109/itsc45102.2020.9294457.

[9] L. Oneto *et al.*, “Dynamic delay predictions for large-scale railway networks: deep and shallow extreme learning machines tuned via thresholdout,” *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 47, no. 10, pp. 2754–2767, Oct. 2017, doi: 10.1109/tsmc.2017.2693209.

[10] C. Wen, W.-W. Mou, P. Huang, and Z. Li, “A predictive model of train delays on a railway line,” *Journal of Forecasting*, vol. 39, no. 3, pp. 470–488, Dec. 2019, doi: 10.1002/for.2639

[11] P. Huang, C. Wen, L. Fu, Q. Peng, and Y.-X. Tang, “A deep learning approach for multi-attribute data: A study of train delay prediction in railway systems,” *Information Sciences*, vol. 516, pp. 234–253, Apr. 2020, doi: 10.1016/j.ins.2019.12.053.

[12] D. Li, W. Daamen, and O. Cats, “Estimation of train dwell time at short stops based on track occupation event data: A study at a Dutch railway station,” *Journal of Advanced Transportation*, vol. 50, no. 5, pp. 877–896, Apr. 2016, doi: 10.1002/atr.1380.

[13] M. E. Gorman, “Statistical estimation of railroad congestion delay,” *Transportation Research Part E-logistics and Transportation Review*, vol. 45, no. 3, pp. 446–456, May 2009, doi: 10.1016/j.tre.2008.08.004

[14] P. Huang, C. Wen, L. Fu, Q. Peng, and Z. Li, “A hybrid model to improve the train running time prediction ability during high-speed railway disruptions,” *Safety Science*, vol. 122, p. 104510, Feb. 2020, doi: 10.1016/j.ssci.2019.104510

[15] W. Barbour, J. O. Mori, S. Kuppa, and D. B. Work, “Prediction of arrival times of freight traffic on US railroads using support vector regression,” *Transportation Research Part C-emerging Technologies*, vol. 93, pp. 211–227, Aug. 2018, doi: 10.1016/j.trc.2018.05.019.

[16] K. Tiong, Z. Ma, and C.-W. Palmqvist, “A review of data-driven approaches to predict train delays,” *Transportation Research Part C-emerging Technologies*, vol. 148, p. 104027, Mar. 2023, doi: 10.1016/j.trc.2023.104027.

[17] G. M. James, D. Witten, T. Hastie, and R. Tibshirani, *An introduction to statistical learning*. Springer International Publishing, 2013. doi: 10.1007/978-1-4614-7138-7.

8. Appendices

Github : [ADS-thesis/ at main · parisasafi/ADS-thesis · GitHub]

▾ 1.2 A merged data set was used to determine the turnover time

```
merged_data = pd.merge(df_3, df_1, left_on=['SL_OMNUMMERING'], right_on=['bewegingcode'], h
merged_data = merged_data[merged_data["SL_VERKEERSDATUM"] == merged_data["verkeersdatum"]]
merged_data = merged_data[merged_data["SL_DRGLPT"] == merged_data["act_drglptVolgend"]]
merged_data = merged_data[merged_data["act_actSoortVolgend"] == "A"]
```

▾ 1.2.1 Convert the time to the right format

```
[ ] merged_data["act_OplantijdVolgend"] = pd.to_datetime(merged_data["act_OplantijdVolgend"], f
    merged_data["act_OplantijdVolgend"]
```

```
4           2022-10-01 07:55:00
1201        2022-10-01 08:55:00
2546        2022-10-01 08:42:00
3287        2022-10-01 09:55:00
4340        2022-10-01 10:55:00
                    ...
181840922   2022-12-10 02:13:00
181840942   2022-12-10 03:04:00
181840962   2022-12-10 03:30:00
181840983   2022-12-10 02:14:00
181841004   2022-12-10 02:52:00
Name: act_OplantijdVolgend, Length: 157650, dtype: datetime64[ns]
```

```
[ ] merged_data["SL_OPLANTIJD_VERTREK"]  = pd.to_datetime(merged_data["SL_OPLANTIJD_VERTREK"],
    merged_data["SL_OPLANTIJD_VERTREK"]
```

```
4           2022-10-01 08:05:00
```

```
[ ] import statsmodels.api as sm
```

```
[ ] OS_model = sm.OLS(y_train, x_train)
    results = OS_model.fit()
```

```
[ ] print(results.summary())
```

```
                                 OLS Regression Results
=======================================================================================
Dep. Variable:     turn over time_seconds   R-squared (uncentered):                   0.617
Model:                                OLS   Adj. R-squared (uncentered):              0.617
Method:                     Least Squares   F-statistic:                          3.368e+04
Date:                    Wed, 12 Jul 2023   Prob (F-statistic):                        0.00
Time:                            09:25:51   Log-Likelihood:                     -8.3348e+05
No. Observations:                  104461   AIC:                                  1.667e+06
Df Residuals:                      104456   BIC:                                  1.667e+06
Df Model:                               5
Covariance Type:                nonrobust
=======================================================================================
                      coef    std err          t      P>|t|      [0.025      0.975]
---------------------------------------------------------------------------------------
SL_MAT_LENGTE       3.6827      0.076     48.300      0.000       3.533       3.832
Wisselmachinist   227.4806      4.301     52.893      0.000     219.051     235.910
matSoort           87.5490      1.046     83.672      0.000      85.498      89.600
aantal_bakken      18.2513      1.600     11.406      0.000      15.115      21.388
spitstype         -96.2763      3.292    -29.249      0.000    -102.728     -89.825
==============================================================================
Omnibus:                   170824.274   Durbin-Watson:                   1.996
Prob(Omnibus):                  0.000   Jarque-Bera (JB):        982194312.280
Skew:                          10.067   Prob(JB):                         0.00
Kurtosis:                     477.610   Cond. No.                         246.
==============================================================================
```

```
[ ] # Splitting the dataset into the Training set and Test set
    from sklearn.model_selection import train_test_split
    x_train, x_test, y_train, y_test = train_test_split(x, y, test_size = 1/3, random_state = 0
```

```
[ ] x_train
```

	SL_MAT_LENGTE	Wisselmachinist	matSoort	aantal_bakken	spitstype
42627	1.002378	1.047361	-1.901712	0.155872	-0.517054
63061	-0.163826	1.047361	0.013453	-0.776460	-0.517054
65892	-1.576726	-0.954775	1.928618	-1.708792	-0.517054
155348	-0.791782	-0.954775	1.290229	-0.776460	-0.517054
14336	-0.343242	-0.954775	-0.624935	0.155872	-0.517054
...	...	...	...	...	...
97653	0.800535	1.047361	0.651841	1.088203	2.444303
95953	-0.343242	-0.954775	-0.624935	0.155872	-0.517054
152332	0.800535	1.047361	0.651841	1.088203	-0.517054
117967	-1.195468	-0.954775	1.290229	-1.242626	-0.517054
43571	-1.262749	1.047361	0.651841	-1.242626	-0.517054

104461 rows × 5 columns

```
[ ] # Fitting Multiple Linear Regression to the Training set
    from sklearn.linear_model import LinearRegression
    regressor = LinearRegression()
    regressor.fit (x_train, y_train)
```

```
▾ LinearRegression
LinearRegression()
```

```
[ ]
```

```
y_pred = regressor.predict(x_test)
#np.set_printoptions(precision=2)
#print(np.concatenate((y_pred.reshape(len(y_pred),1), y_test.reshape(len(y_test),1)),1))
from sklearn.metrics import mean_absolute_error, r2_score
mae = mean_absolute_error(y_test, y_pred)
print("MAE:", mae)

# Calculate R-squared (R2) score
r2 = r2_score(y_test, y_pred)
print("R2 Score:", r2)
```

```
MAE: 425.9748666756712
R2 Score: 0.025675471248122417
```

```
[ ] mse = np.mean((y_test - y_pred) ** 2)
```

```
/usr/local/lib/python3.10/dist-packages/numpy/core/fromnumeric.py:3472: FutureWarning: In a
  return mean(axis=axis, dtype=dtype, out=out, **kwargs)
```

```
cv_scores = []
alpha_values = np.logspace(-4, 0, num=10)

for alpha in alpha_values:

    lasso = Lasso(alpha=alpha)

    scores = cross_val_score(lasso, x_train, y_train, cv=5, scoring='neg_mean_squared_error

    cv_scores.append((-1) * scores.mean())  # Convert negative MSE to positive


# Select the best alpha

best_alpha = alpha_values[np.argmin(cv_scores)]


# Train the final model using the best alpha

lasso = Lasso(alpha=best_alpha)

lasso.fit(x_train, y_train)
```

```
▾ Lasso
Lasso(alpha=0.0001)
```

```
[ ] best_alpha
```

```
0.0001
```