Short Term Delay Prediction in Passenger Railways

Using Machine Learning
Applied in the Dutch Rail Network

Eva Lehká

Short Term Delay Prediction in Passenger Railways

Using Machine Learning; Applied in the Dutch Rail Network

E. Lehká

December 2019

in partial fulfilment of the requirements for the degree of

Master of Science

in Transport, Infrastructure and Logistics

at the Delft University of Technology

Faculty of Civil Engineering & Geosciences

Faculty of Mechanical, Maritime and Materials Engineering

Evaluation committee:

Prof. Dr. Ir. Serge P. Hoogendoorn	TU Delft	Chairman
Dr. Ir. Niels van Oort	TU Delft	Supervisor
Ir. Mark Duinkerken	TU Delft	Supervisor
Pieter-Jan Fioole	Nederlandse Spoorwegen	Supervisor

Preface and acknowledgments

People like to ask others about their plans for the future. Some may have a clear idea and know exactly what they want to do. That is not my case. Instead, I am open to any turn in my life that feels right. This way, I got to study transportation and eventually I found myself at the Technical University in Delft. Although it was not always an easy path, I am very grateful I had the chance to study there and I am especially grateful that I got the opportunity to do an internship at NS as a part of my graduation. Despite, or maybe thanks to, the difficulties I encountered, it was an incredible and enriching experience. Not only because I gained insight into a new field, machine learning and specifically decision tree ensemble methods. I also had a chance to be a part of an amazing team 'PI', to whom I would like to say thank you for the warm and welcoming atmosphere I was accepted into. Furthermore, I could not even start and certainly finish my thesis without guidance, advices and academic and moral support I received from my supervision committee, who I would like to thank as well.

I would like to thank all the people that made my stay in Delft possible and thanks to whom I had such a great time there. At the first place, it is my parents and my brother whose support to me cannot be described by words. Next, it is my boyfriend who endured even the most stressful times with me and always showed up with chocolate and a glass of wine at the right moment. It is also my close friends who always cheered me up and ensured me that I was doing fine. A special thanks also belongs to my colleagues at a three-dotted pizza restaurant chain who scheduled my working hours to fit them perfectly in my academic agenda. And finally, my thanks goes to OHMies, an amazing yoga group that always uplifted my spirit and kept my body, mind and soul in balance.

Thank you.

Short Term Delay Prediction in Passenger Railways

Using Machine Learning; Applied in the Dutch Rail Network

Eva Lehká – Transport, Infrastructure and Logistics –Delft University of Technology

***Abstract*— We test the effect of a variety of feature sets representing passenger volumes, weather conditions and train interactions, when defined as features and used in a gradient boosting model to predict passenger train delays 20 minutes to the future from the last registration point. Effects of the features and their combinations on the prediction quality are analyzed and the best performing feature sets selected. The results showed that the passenger volumes features (in the form as defined in our work) do not have any prediction power and rather introduced noise in the predictions. The weather features resulted in reduced expected delay change with a slight positive effect on precision of the classification task while worsening the recall. The largest positive effect was observed when train interaction features were introduced despite their highly simplified form. Considering the low computational efforts necessary to retrieve the features, we conclude there is a potential for application of similarly defined train interactions features in other models.**

Keywords*— *Train delays, Machine learning, XGBoost, Gradient Boosting, Weather conditions, Passenger counts, Train interactions

I. Introduction

Our work engages in short term delay prediction in passenger railway, specifically prediction of delays of all regularly scheduled passenger trains. Although delay prevention should be a primary concern. However, prior knowledge of upcoming delays may mitigate inconvenience induced to the passengers by providing them with an opportunity to change their itineraries, or possibly might even eventually help to prevent delay propagation by prompt dispatching interventions.

The objective of our research is to define and evaluate sound feature sets reflecting effects of passenger counts, weather conditions and train interactions on passenger trains delays, which are to be used along with features derived from data available in the RAS competition (INFORMS, 2018), on which our research builds on, to forecast passenger trains' delays 20 minutes to the future. Effects of the individual features and their combinations are reflected upon. Note that the span of 20 minutes to the future refers to the last registration point within a 20-minutes time window since the last registration point.

For this purpose, we use a decision trees based gradient boosting method XGBoost (T. Chen & Guestrin, 2016). Selection of the method is based on the conclusions drawn by Van den Bulk et al. (2018), whose work served as a starting point of our research.

Our research has been conducted in cooperation with Nederlandse Spoorwegen, commonly referred to as NS, the principal passenger railway operator in the Netherlands who provided us with data, resources and valuable knowledge to carry out our study, and to test and assess our models on the Dutch passenger railway network. Besides NS, valued support was accommodated also by ProRail, a rail infrastructure manager in the Netherlands, who contributed by providing a substantial part of the data we used in our work.

--- structure of the paper ----

II. Motivation and prior work

The most commonly used factors in delay prediction in railways are network- and timetable-based such as departure and arrival times, location identification, event type, train identification and similar (Hansen, Goverde, & Van Der Meer, 2010; Kecman, 2014; Kecman & Goverde, 2015a; Lessan, Fu, & Wen, 2019; Nair et al., 2019; Oneto et al., 2016; Wang & Work, 2015; Yaghini, Khoshraftar, & Seyedabadi, 2013). Similar attention has been paid to delay propagation and its prediction (Berger, Gebhardt, Müller-Hannemann, & Ostrowski, 2011; Corman & Kecman, 2018; Goverde, 2010; Peters, Emig, Jung, & Schmidt, 2006; Jianxin Yuan, 2007). However, in contrary to the lengthy list of publications considering the earlier factors in train delay prediction models, weather conditions were considered in much fewer models (Nair et al., 2019; Oneto et al., 2016) with a support of an observed correlation between weather conditions and train delays (Ling, Peng, Sun, Li, & Wang, 2018). Passenger counts and the effects of crowding attract a lot of attention among transport researchers. In general, especially dwell time (Kecman & Goverde, 2015b; San & Mohd Masirin, 2016) and psychological effects of crowding (Cox, Houdmont, & Griffiths, 2006; Tirachini, Hensher, & Rose, 2013) are in the spotlight. However, application of the passenger counts and crowding in the railway delay prediction is rather scarce (Albert, Kraus, Müller, & Schöbel, 2017), though some examples can be found in the bus transit domain (M. Chen, Liu, Xia, & Chien, 2004; M. Chen, Yaw, Chien, & Liu, 2007).

All the selected factors, passenger counts, weather and train interactions (in the meaning of delay propagation) were somehow implemented in some delay prediction model. However, none of the publications reviewed presented an attempt to include all the factors into one model. Furthermore, many types of models were presented in the reviewed publications such as stochastic models (Berger et al., 2011; Huisman & Boucherie, 2001; J Yuan, Goverde, & Hansen, 2002), Bayesian networks (Corman & Kecman, 2018; Lessan et al., 2019), data mining techniques (Hansen et al., 2010) or Neural network (Peters et al., 2006; Yaghini et al., 2013; Oneto et al., 2016). Decision trees based models appeared mainly in the form of a random forest model (Oneto et al., 2016; Nair et al., 2019). No application of gradient boosting model, such as XGBoost, was found in the context of train delay prediction, and especially not in a combination with all the factors affecting delays as mentioned above. Finally, most of the publications that were offering any comparison of multiple models focused on assessment of the differences in the models' architecture rather than comparing an impact of various factors considered in the models. The consequent research gap was found in the following three aspects: 1) Application of a gradient boosting decision trees-based model, specifically XGBoost, 2) Inclusion of passenger counts data into a passenger trains delay prediction model, 3) Combination of timetable-based information, weather condition, passenger counts and train interactions for delay prediction, implementing each factor separately and in various combinations with the other factors.

III. Model

We develop a number of models to observe and assess effects of a variety of features described in Section A. Although the prediction is intended for the whole network (all regularly scheduled passenger trains operated by NS in the Netherlands) the models are build for every train series individually. That is argued by an assumption that there are certain conditions unique for every train series. Therefore, 79 models are made for every feature set. The individual results are shortly reviewed but attention is paid to the overall performance of the combined results.

A. Data and derived features

Input form compatible with the XGBoost model can be described as a vector, i.e. a set of features with varying units and scales. A set of characteristics captured by a vector is referred to as a feature set and individual characteristics (i.e. vector elements) as features. The target variable, train delay in our case, is then referred to as a label. To avoid overfitting, the dataset is split into smaller instances called a training, testing and evaluation set (by proportions of 0.6, 0.2 and 0.2, respectively). The first two sets are used for tuning of the model's configuration and consecutively for the final model training. Eventually, the evaluation set is used for the model's performance assessment. The data from which the features are derived consisted of the timetable, the realization data (automatic vehicle location data), weather data (historical hourly measurements), passenger data (passenger count estimations based on smart card data) and minimum required headways between pairs of train series at specific locations.

The resulting features were divided into a number of feature sub-sets which were then combined into feature sets for the prediction models development. Furthermore, some of the sub-sets were defined in multiple variations. The features belonging into the categories along with the number of variations created (in the brackets) are as follows:

- *Basic features (1):* Day of the week, Hour, Minute, Location, Direction, Current delay, Delay 1 before, Delay 2 before.

- *Locations features (2)*: Number of the activities to come or their frequency in the timetable: V (departure), K_V (Short departure), D (pass through).

- *Passengers features (5)*: Seats ratio, Peak departure, Total number of p.*, Boarding p.*, Alighting p.* (*Scaled to the median number of passengers for given spatial-temporal criteria).

- *Weather features (3)*: Average wind speed, Highest wind speed, Temperature, Precipitation, View distance, Mist, Snow, Rain, Storm, Ice, Bad weather.

- *Interactions features (6)*: Expected headway, Expected violated headway, Binary interaction identification. All either per train series or train type category.

The feature sub-sets were gradually combined according to the scheme in Table I.

TABLE I. Testing scheme of combination of the feature sets and their variations

Step	Feature sets	Number of models
1	Base	1
2	Base, Locations	2
3	Base, Locations, Passengers	5
4	Base, Locations, Weather	3
5	Base, Locations, Interactions	6
7	Base, Locations, Passengers, Weather, Interactions	1

B. XGBoost

The selected booster parameter was '*gbtree'* (a tree-based model), the task was regression, the objective was set to '*reg:squarederror*' (regression: squared error) and the evaluation metric was '*rmse*' (root mean square error). Hyperparameters, i.e. parameters defining rules for the tree-building process, require tuning to fit the specific problem. Therefore, they were tuned for every feature set category. A group of the best performing hyperparameter sets was used as candidates for the following individual models, which then selected the best fitting setting out of the candidates. The tuned hyperparameters and the allowed values are presented in Table II.

TABLE II. XGBOOST HYPERPARAMETERS AVAILABLE FOR TUNING.

Parameter	Possible values
learning rate	[0.001; 0.01; 0.1]
n estimator	[500; 750; 1000; 1250]
max depth	[5; 6; 7; 8; 9; 10]
child weight	[1; 2; 3; 4; 5]
gamma	[0; 0.1; 0.2; 0.3; 0.4]
alpha	[0; 0.01; 0.1]
subsample	0.8
colsample_bytree	0.8

C. Key performance indicators

Performance of the predictions was evaluated by multiple performance indicators.

- *MSE and Confidence intervals, RMSE, RWMSE.*

MSE and RMSE are calculated as in (3.1) and (3.2) respectively, where N is the number of observations, is a vector of the forecasted values and is a vector of the observed values. In the context of our research, the unit of RMSE is minutes. The MSE is complemented by its corresponding 95% confidence interval (CI). To determine the CI of the MSE, the normal distribution of the squared errors is required. A normality test defined as , where s is the z_score of a skew test, and k is the z_score of a kurtosis test (Bai & Ng, 2005), is used. The hypothesis that the sample is normally distributed is rejected if p < 0.05 (Biau, Jolles, & Porcher, 2010). In case the hypothesis is rejected, the data is resampled using a bootstrap method (Hesterberg, 2015; Kesar Singh and Minge Xie, n.d.; Pek, Wong, & Wong, 2017). An algorithm to obtain the bootstrapped MSE along with its 95% confidence intervals was adopted from Good (2006).

$$MSE = \frac{\sum_{i=1}^{N}\left(z_{f_i} - z_{o_i}\right)^2}{N} \qquad (3.1)$$

$$RMSE = \sqrt{MSE} \qquad (3.2)$$

In addition, root weighted mean square error (RWMSE) was adopted from the assessment indicators used in the RAS competition retrieved from the submitted work of Nabian et al. (2018). The RWMSE is defined as in (3.3) where there is an additional variable, the weight wi defined in (3.4) which gives more importance to errors associated with larger observed delays.

$$RWMSE = \sqrt{\frac{\sum_{i=1}^{N} w_i\left(z_{f_i} - z_{o_i}\right)^2}{N}} \qquad (3.3)$$

$$w_i = \begin{cases} 0.2 & if\ \ z_{o_i} \in [-1{,}1] \\ 0.8 & if\ \ otherwise \end{cases} \qquad (3.4)$$

- *Confusion matrix.*

The confusion matrix represents number of instances that fall into bins of combinations of predicted and actual (true) values. The diagonal then represents correctly predicted classes. The classification tasks used in our research can be described as binary and multi-class classification. In both cases, one and only one class is predicted for each instance. The sum across the matrix therefore equals to the number of instances that were subject to the prediction (Sokolova & Lapalme, 2009).

- *Accuracy, Precision, Recall and F1 Score.*

Correctness of the predictions in classification tasks can be represented by accuracy, precision, recall and F1-score. Accuracy equals to the percentage of correctly predicted classes (3.5), therefore a percentage of the sum on the diagonal of a confusion matrix.

$$Accuracy = \frac{\sum_{i=1}^{n} N_{ii}}{\sum_{i=1}^{n}\sum_{j=1}^{n} N_{ij}} \qquad (3.5)$$

As accuracy may be misleading when representation of the classes is strongly disbalanced, attention is paid to the remaining metrics instead. Precision represents an accuracy given a certain predicted class (3.6). Opposite to that, recall is an accuracy supposing a specific actually observed class (3.7).

$$Precision_i = \frac{N_{ii}}{\sum_{k=1}^{n} N_{ki}} \qquad (3.6)$$

$$Recall_i = \frac{N_{ii}}{\sum_{k=1}^{n} N_{ik}} \qquad (3.7)$$

A harmonic mean of the two metrics, precision and recall, is called F-score, or commonly F1-score, where 1 denotes that no weighting is used and precision and recall both have the same weight in the equation (3.10).

$$F1\_score_i = \frac{2 \cdot Precision_i \cdot Recall_i}{Precision_i + Recall_i} \qquad (3.8)$$

- *Feature importance: Gain.*

A feature importance indicator gain (is an "average gain across all splits the feature is used in" (XGBoost Developers, 2016c). Given a feature set F and a set of trees T, each tree consists of a set of splits S_t and is used by a number of observations O_t , where each observation follows a sequence of splits K_o^t. Gain then can be mathematically defined as in (3.8).

$$gain_f = \frac{\sum_{t=1}^{T}\sum_{o=1}^{O_t}\sum_{k=1}^{K_o^t} split_{t,o}^{k} \cdot \left(loss_{t,o}^{k-1} - loss_{t,o}^{k}\right)}{\sum_{t=1}^{T}\sum_{o=1}^{O_t}\sum_{k=1}^{K_o^t} split_{t,o}^{k}} \qquad (3.8)$$

$$(\ f \in F, split_{t,o}^{k} \in [0{,}1], loss_k \in \mathbb{R}_+\ )$$

where $loss_{t,o}^{k}$ is a value given by a loss function after introducing a $split_{t,o}^{k}$. $loss_{t,o}^{0}$ then refers to the loss function value before any split is introduced.

The value of gain is calculated for every individual model, i.e. for every train series, and the absolute values are not directly comparable. Therefore, we compare the individual gains to the highest gain of the individual model to obtain a relative value which is then comparable among the train series. Furthermore, for the final evaluation, the average relative gain among the train series is calculated, which is the value presented in this work.

IV. Results

The results are presented by the key performance indicators presented earlier. Therefore, the models' performance is compared by the RMSE and related metrics, by the scores resulting from confusion matrices, and eventually, features' importance is inspected. As performance of the models built on the diverse feature sets considerably varies among the train series, the best performing model with respect to the RMSE was selected for each train series, the resulting predictions were combined across all the train series and evaluated as if each train series was predicted with the best fitting feature set. These results are referred to as 'All combined'.

Note that the feature set categories will be identified by following abbreviations: B (Base), L (Locations), P (Passengers), W (Weather) and I (Interactions). Versions of the feature sub-sets within the feature categories then are identified by a number.

A. Naïve forecast: The reference

The Naïve prediction was computed as strictly subtracting 1 minute from the 'current' delay. However, prediction by the model was done to the last train registration within a 20 minutes time window since the last registration point.

B. Regression

Basic assessment of the regression results is done by comparison of the RMSE's. All the results are presented in the Table III along with the corresponding MSE's and the 95% confidence intervals. Focusing on the RMSE, results of the B0 feature set correspond with our expectations as the RMSE of 1.52 outperforms the Naïve forecast's RMSE of 1.99 but not the results of the best performing model from the research by Van den Bulk et al. (2018), which reached a RMSE of 1.34 and which included further features in comparison to our basic feature set. The lowest overall value (RMSE 1.48) is reached with the feature sets that include train interactions either in the form of expected headway (B0L1I3) or expected violated headway (B0L1I4). The latter one shows a higher upper bound of the confidence interval by 0.01. Nevertheless, both reach the lowest RMSE by the most train series (29% and 16% respectively). The remaining train series reach the best performance with feature set B0L1W1 (10% of train series), B0L1 (9% of train series) followed by B0L1W2, B0L1I5 (both 6% of train series) and B0L1W0 (5% train series). Within the subset of 27 train series that were used for comparison of the feature sets containing all versions of train interactions feature sub-sets, the feature sets containing features per train type surprisingly outperformed their more complex alternatives in both, the overall RMSE and the number of train series performing the best when the respective feature set is used.

Next to the generic RMSE, a RWMSE was defined to emphasize errors associated with large observed delays. See the definition in (3.3) and (3.4). The RWMSE was calculated on the feature set level only. Assessing the feature sets based on this metrics, the differences become significantly smaller. The best performance is observed on the feature sets B0L1, B0L1I3-5 with RWMSE of 1.35, followed by B0L0 and B0L1P0 with RWMSE of 1.36. The remaining feature sets reach RWMSE of 1.37 with an exception of the B0 feature set that is burdened by a RWMSE of 1.38.

Based on the RMSE and the RWMSE, the best performing versions of the feature set categories were selected. For the Passenger category, it was the version 0 and so it was for the Weather category. In selection of the best performing version of the feature set containing the Interactions category, the number of best performing train series was considered in addition to the RMSE and the RWMSE. Thus, the version 3 was selected. Selection of the versions is in line with performance of the individual train series within the feature categories as the most train series reach their best results in the categories with the sub-set versions L1, P0 and W0 respectively. Altogether, the final feature set can be denoted as B0L1P0W0I3. This feature set was created despite the poor performance of some of the feature sub-sets in the simpler feature sets. As can be observe, the RMSE and RWMSE are both worse than was the case of B0L1 and B0L1I3-5.

Finally, see the lower part of the Table III for comparison of all the six feature set versions including Interactions features. Note that the prediction was made for a subset of 27 train series due to computational power limitations we encountered and thus the results cannot be directly compared to the results in the upper part. Instead, the results provide a comparison among the versions of the train interactions feature sets. Contrary to our expectations, looking at the RMSE, the overall results are slightly worse when the interactions are defined per individual train series (versions 0-2) instead of train types (versions 3-5).

The difference between the feature sets diminishes when the RWMSE is used as an evaluation measure. Nevertheless, the binary encoded Interactions features in the B0L1I2, and the corresponding simplified version B0L1I5 are still clearly the least effective.

TABLE III. REGRESSION RESULTS.

B	L	P	W	I	RMSE	MSE	95% CI	RWMSE
Naïve					1.99	3.94	[3.90, 3.98]	1.91
0					1.52	2.30	[2.27, 2.33]	1.38
0	0				1.50	2.24	[2.21, 2.27]	1.36
0	1				1.49	2.23	[2.20, 2.27]	1.35
0	1	0			1.50	2.26	[2.23, 2.29]	1.36
0	1	1			1.50	2.26	[2.23, 2.29]	1.37
0	1	2			1.51	2.27	[2.24, 2.30]	1.37
0	1	3			1.51	2.27	[2.23, 2.31]	1.37
0	1	4			1.51	2.27	[2.24, 2.30]	1.37
0	1		0		1.50	2.26	[2.23, 2.30]	1.37
0	1		1		1.51	2.27	[2.24, 2.30]	1.37
0	1		2		1.51	2.27	[2.23, 2.30]	1.37
0	1			3	1.48	2.20	[2.17, 2.23]	1.35
0	1			4	1.48	2.20	[2.17, 2.24]	1.35
0	1			5	1.49	2.22	[2.19, 2.26]	1.35
0	1	0	0	3	1.50	2.25	[2.21, 2.28]	1.36
0	1			0	1.56	2.43	[2.38, 2.47]	1.41
0	1			1	1.56	2.43	[2.38, 2.47]	1.41
0	1			2	1.57	2.45	[2.40, 2.49]	1.42
0	1			3	1.55	2.41	[2.37, 2.46]	1.41
0	1			4	1.55	2.41	[2.37, 2.46]	1.41
0	1			5	1.56	2.44	[2.40, 2.49]	1.42
All combined					1.48	2.20	[2.16, 2.23]	1.34

C. *Derived classification tasks*

Three classification tasks are defined and will be referred to as follows:

- *Delay existence*: Positive class (a delay exceeding 2 minutes is predicted), negative class (a not existing delay or a delay of maximum 2 minutes is predicted)
- *Delay jump*: Positive class (a delay jump of 4 minutes or more is predicted), negative class (a not existing delay jump is predicted)
- *Delay change*: Decreasing class ('-'; a delay is predicted to decrease by 2 or more minutes), Constant class (a delay is predicted not to change by 2 or more minutes in either direction), Increasing class ('+'; a delay is predicted to increase by 2 or more minutes)

Representation of the classes is significantly disbalanced in our case as, for example, low delays predominating high delays and small delay changes predominating large delay changes. Reaching high scores in such highly frequented class is not any surprise and attention should be paid to scores of the less frequented class(es).

Delay existence.

With respect to performance in predicting whether a delay will take place or not, the best performing feature sets are B0L1I3 and B0L1I4 if we look at the F1 score. Also the 'All combined' model reaches equal F1 score. Those are followed by B0L1P0W0I3, B0L1, and B0L0 and B0L1I5 in this order. An interesting observation is that weather feature sets lead to high scores in precision and lower scores in recall. Passenger features then show the opposite tendency. Precision in this context represents how many of the instances predicted to be delayed were also observed to be delayed. Low score in precision thus suggests that the delays tend to be overestimated. That is what the features reflecting passenger volumes seem to cause because the precision scores are significantly worsened compared to the simpler feature sets. Other than that, the precision scores are relatively equal among the other feature sets. As mentioned, recall noticeably drops when weather features are included (B0L1W0-2). That suggests delays are rather underestimated and thus some of the delays fall below the threshold of 2 minutes. Interestingly, low recall score is observed also with the B0L1P0W0I3 feature set, while it reaches the overall highest score in precision. High score in precision is achieved also by the 'All combined' model. See an overview of the scores in Table IV.

TABLE IV. DELAY EXISTENCE PREDICTION RESULTS.

Feature sets					F1	Precision	Recall	Accuracy
B	L	P	W	I				
Naïve forecast					0.586	0.462	0.798	0.898
0					0.655	0.548	0.814	0.909
0	0				0.659	0.551	0.819	0.911
0	1				0.660	0.553	0.819	0.911
0	1	0			0.657	0.552	0.812	0.910
0	1	1			0.654	0.545	0.816	0.910
0	1	2			0.653	0.545	0.815	0.910
0	1	3			0.654	0.546	0.815	0.910
0	1	4			0.653	0.545	0.815	0.910
0	1		0		0.656	0.559	0.795	0.909
0	1		1		0.655	0.558	0.793	0.908
0	1		2		0.655	0.558	0.793	0.908
0	1			3	0.665	0.559	0.820	0.912
0	1			4	0.665	0.559	0.820	0.912
0	1			5	0.659	0.550	0.822	0.911
0	1	0	0	3	0.661	0.564	0.799	0.910
All combined					0.665	0.561	0.816	0.912
0	1			0	0.655	0.552	0.806	0.904
0	1			1	0.655	0.551	0.808	0.904
0	1			2	0.647	0.537	0.813	0.903
0	1			3	0.656	0.549	0.813	0.904
0	1			4	0.656	0.549	0.813	0.904
0	1			5	0.648	0.537	0.815	0.903

Comparing train interactions feature set variations with respect to the F1 score, the more complex versions (B0L1I0-2) perform worse than the simpler ones (B0L1I3-5). Also comparison of recall scores suggests the same conclusion. On the other hand, precision is higher when B0L1I0-1 feature sets are used. The feature sets B0L1I2 and B0L1I5 perform worse than the alternative feature sets with an exemption of recall where they perform better than the other versions.

Delay change.

Expectedly, a more challenging task is to predict in which direction a delay will change which is confirmed by the generally lower scores. Because we derived the delay change classification from the regression results, we did not predict directly the delay change. Instead, the classes represent whether the difference between the predicted and the present delay corresponds with the observed change. Referring to Table V, decrease of delay occurs in 6.3% of the available observations and increase in 7%. The remaining 86.7% of observations show change of delay smaller than 2 minutes (in either direction). Therefore, there is a clear predominance of the 'constant class', which thus understandably scores well in this classification problem (with F1 scores around 0.93, precision 0.98 and recall approximately 0.88).

Low precision of a class implies decrease in recall of another class or classes. As can be seen in Table V, precision of the 'decreasing' and 'increasing' classes are very low and recall of the 'constant' class is significantly lower than its precision. It is likely, that if an existing delay change (of any direction) is not predicted correctly, it is predicted as 'constant'. That was confirmed by an inspection of the data used for calculation of the metrics presented in Table V. Especially low scores are reached in precision of the 'increasing' class. As observation of the data revealed, falsely predicting 'decreasing' class instead of the correct 'increasing' class is approximately twice as likely than vice versa. On the other hand, false predictions of observed 'constant' class are equally distributed between the two other classes. In general, predicted delay changes exceeding 2 minutes are with high probability actually smaller than 2 minutes, no matter the direction of the change. That means, the predictions appear to overestimate delay changes. Predicted delay increase is correctly predicted only in approximately 13-16% cases and delay decrease in approximately 31-34% of the cases. Contrary to that, recall scores are relatively high, showing that decrease of delay is correctly predicted in approximately 67-70% and delay increase in about 49-58%.

TABLE V. DELAY CHANGE PREDICTION RESULTS.

Feature sets					F1		Precision		Recall		Accuracy
B	L	P	W	I	-	+	-	+	-	+	
0					0.422	0.217	0.309	0.135	0.667	0.551	0.868
0	0				0.445	0.225	0.328	0.141	0.688	0.567	0.870
0	1				0.453	0.229	0.337	0.143	0.691	0.565	0.871
0	1	0			0.446	0.226	0.331	0.143	0.684	0.537	0.869
0	1	1			0.427	0.215	0.310	0.133	0.687	0.554	0.869
0	1	2			0.428	0.216	0.311	0.134	0.688	0.558	0.869
0	1	3			0.428	0.215	0.311	0.133	0.687	0.553	0.869
0	1	4			0.426	0.216	0.309	0.134	0.686	0.554	0.869
0	1		0		0.457	0.241	0.345	0.159	0.676	0.501	0.868
0	1		1		0.451	0.239	0.339	0.157	0.674	0.497	0.867
0	1		2		0.452	0.240	0.340	0.158	0.675	0.498	0.868
0	1			3	0.451	0.245	0.332	0.155	0.701	0.579	0.872
0	1			4	0.454	0.245	0.336	0.156	0.698	0.575	0.872
0	1			5	0.446	0.225	0.328	0.139	0.699	0.579	0.871
0	1			0	0.427	0.283	0.309	0.188	0.695	0.568	0.859
0	1			1	0.426	0.281	0.307	0.186	0.692	0.568	0.859
0	1			2	0.405	0.253	0.286	0.161	0.696	0.587	0.859
0	1			3	0.417	0.276	0.296	0.180	0.702	0.593	0.860
0	1			4	0.424	0.275	0.304	0.180	0.697	0.586	0.860
0	1			5	0.409	0.253	0.290	0.161	0.696	0.591	0.859
0	1	0	0	3	0.455	0.253	0.342	0.167	0.681	0.521	0.868
All combined					0.460	0.250	0.343	0.160	0.698	0.574	0.871

Looking at differences among the feature sets, the highest F1 scores are reached by B0L1P0W0I3 and the 'All combined' model, followed by feature sets containing train interactions features and weather features. In accordance with the previous classification task, passenger counts features cause delay change overestimation leading to low precision and consequently F1 scores. On the other hand, weather features seem to have a significantly positive effect on delay change direction prediction as the results of B0L1W0-2 is very comparable to the results of B0L1I3-4 which so far outperformed weather features. However, feature sets B0L1W0-2 lag behind in recall, especially in recall of the 'increasing' class, where they perform worse even than B0.

Among the feature sets B0L1I0-5, the F1 and precision scores are surprisingly higher in the more complex feature sets (B0L1I0-2) compared to their simpler alternatives. Conversely, the recall scores are higher when B0L1I3-5 are used and so is the overall accuracy. Predictions by models using the complex feature sets (B0L1I0-2) thus are more correct with respect to delay change identification, while the actual delay changes are correctly predicted with a slightly higher probability by the simpler feature sets (B0L1I3-5).

Delay jump.

TABLE VI. DELAY JUMP PREDICTION RESULTS.

Feature sets					F1	Precision	Recall	Accuracy
B	L	P	W	I				
0					0.072	0.040	0.390	0.982
0	0				0.083	0.046	0.442	0.983
0	1				0.090	0.050	0.449	0.983
0	1	0			0.084	0.047	0.405	0.983
0	1	1			0.072	0.039	0.412	0.983
0	1	2			0.071	0.039	0.411	0.983
0	1	3			0.073	0.040	0.418	0.983
0	1	4			0.071	0.039	0.409	0.983
0	1		0		0.090	0.051	0.379	0.982
0	1		1		0.089	0.050	0.390	0.983
0	1		2		0.086	0.049	0.372	0.982
0	1			3	0.090	0.050	0.470	0.983
0	1			4	0.090	0.050	0.457	0.983
0	1			5	0.087	0.048	0.473	0.983
0	1	0	0	3	0.088	0.050	0.393	0.982
All combined					0.091	0.050	0.476	0.983
0	1			0	0.065	0.036	0.356	0.980
0	1			1	0.063	0.035	0.361	0.980
0	1			2	0.051	0.027	0.352	0.980
0	1			3	0.057	0.031	0.393	0.980
0	1			4	0.052	0.028	0.374	0.980
0	1			5	0.055	0.030	0.373	0.980

Delay jump is a delay change equal to or exceeding 4 minutes regardless direction of the change. According to that, delay jump can be observed in less than 3% of instances in the full dataset. That implies considerable disproportionality to the opposing class which reaches equal scores across all the feature sets (F1 score 0.991, precision 0.999 and recall 0.984). Performance of the prediction with respect to this classification naturally cannot outperform combined prediction of the 'decreasing' and 'increasing' class above. Precision scores are therefore unsurprisingly low, predicting a delay jump correctly only in approximately 4-5% of the cases a delay jump is predicted. On the other hand, existing delay jumps are correctly predicted in about 37-47% of the cases. That means, delay jumps are highly overestimated, or in other words, delay changes are predicted to excess the threshold of a 4-minutes change. Inspection of the results revealed that approximately 10 times more delay jumps were predicted than were actually observed.

Accordingly to the results of the 'delay change' task, the highest F1 scores are reached by feature sets B0L1, B0L1I3-4 and B0L1W0. The latter feature set reaches the highest precision score, however, lags behind in recall and the overall accuracy. The most complex feature set B0L1P0W0I3 does not reach the highest performance especially due to poor recall. Contrary to that, the very highest recall score is attained by the 'All combined' model what contributes to the highest measured F1 score.

D. *Feature importance*

Throughout the results, the present delay ('Delay') clearly reaches the highest scores in gain between 0.97 and 1.0 which means that the feature brings the most gain in a vast majority of the individual models, and especially when more features are present. In the B0 feature set, the leading feature is the direction ('Direction'), and in the B0L0-1 feature set, the features with a considerably high gain were also the hour ('Hour'), the upcoming departure locations feature ('L0 / L1: V') and the upcoming pass-through locations feature ('L0 / L1: D').

The remaining features have a rather supplementary role in the gain maximization. The next feature with relatively high scores in gain is the previous delay ('Delay_1before') scoring between 0.31 and 0.40. The following features such as the direction, a delay two locations back ('Delay_2before') and upcoming locations identification ('L0 / L1: V', 'L0 / L1: K_V', 'L0 / L1: D') then score between 0.1 and 0.2. However, when passenger counts or weather conditions are introduced, scores of the upcoming locations identification drop to or below 0.1. Train interactions do not cause such a drop. Nonetheless, the features of the B0 feature set show relatively stable importance across all the feature sets combinations. That is also due to the fact that it is always the present delay that reaches the highest gain and the remaining features of the set always score proportionally approximately equally.

Assessing the additional features, the upcoming locations identification features score relatively high (between 0.08 and 0.14) compared to the other features. Slightly higher scores are observed on the upcoming departure locations ('L0 / L1: V') followed by the upcoming pass-through locations features ('L0 / L1: D'). All the features reflecting passenger counts ('Pax_Total_1-4', 'Pax_Board_1-4', 'Pax_Alight_1-4'), the seats ratio ('seats_ratio') or the peak hour ('peak_dep') score comparably (0.04 or 0.05) although the relative gain of the peak hour decreases when the passenger counts features are included. That is understandable as the peak hour is partially implicit in the passenger counts features. The relative average gain of the features representing weather conditions resembles the scores of the passengers-related features varying between 0.03 and 0.06. The highest score of 0.06 is obtained by the precipitation ('Precp'), temperature ('Temp') followed by the wind, view distance, rain and bad weather identification ('Avgwind', 'Highwind', 'View', 'Rain', 'BadWeather') scoring 0.05, and by the snow, storm, ice, and mist features

(‘Snow’, ‘Storm’, ‘Ice’, ‘Mist’) scoring 0.04 or 0.03. Finally, the average relative gain of the train interactions features shows that the features representing interactions with empty rolling stock (‘LM’) are utterly powerless (scores of 0.00), and that identification of interactions with IC trains (‘IC’) scores relatively higher than interactions with SPR trains (‘SPR’) by scores of 0.04 or 0.05 compared to 0.03 and 0.04 respectively.

V. Results summary

The highest improvement of performance comes from the basic feature set (B0), which, compared to the Naïve forecast, decreases the RMSE from 1.99 to 1.52, and decreased width of the MSE 95% confidence interval by 25%. The features representing the upcoming locations in the feature sub-sets L0 and L1 further decreased the RMSE by 0.01 and 0.02 respectively (RMSE 1.50 and 1.49). The passengers features added to the B0L1 feature set caused increase of the RMSE by 0.01 or 0.02 and so did the weather features. The train interactions features on the other hand caused an improvement by decreasing the RMSE by 0.01 (RMSE 1.48) except for the last version (I5) which did not change the RMSE but shifted the MSE 95% confidence interval by -0.01. Comparing differences between all the train interactions (tested on a subset of train series), the lowest RMSE is reached with the version I3 and I4, followed by I5 with the RMSE higher by 0.01, I0 and I1 with an additional increase of the RMSE by 0.01 and finally I2 with another increase by 0.01 on the RMSE. Finally, the feature set consisting of feature sub-sets of all the categories is burdened by a RMSE of 1.50. Results of the underlying individual models built for the separate train series largely vary among the feature sets and train series and every feature set performs the best for at least one train series. Performance of the feature sets with respect to the remaining performance indicators follows relatively the same pattern. However, weather features bring an improvement in delay change and delay jump prediction, especially in the prediction precision.

The features were assessed by a feature importance metrics called gain, which was then scaled to the highest observed value in the respective model. The present delay reached the highest scores in gain in a vast majority of the individual models. Next highly scoring features were the direction, hour, the upcoming departure locations and the upcoming pass-through locations feature and the previous delays. However, considering the predictions’ performance results, the feature sets including train interactions features achieved the highest scores and thus the features must have a significant and positive impact on the prediction. Comparison of performance among the respective feature set variations, the binary and the derived quasi-binary versions (sum of the underlying binary features) are significantly outperformed by those representing expected headways and expected violated headways. Preference between those two is not so straightforward and the differences are rather negligible although with a slight preference towards the first version. Comparison of the features being defined per train category or train series also shows rather small nuances, however, the latter case features have a positive impact on delay jump prediction which is a viable argument in their favor. Finally, the weather features showed a significant positive impact on precision of prediction of delay change direction and delay jump. That means, the features inclusion resulted in larger expected delay changes, which on the other hand lead to overestimation of the delay changes and decreased recall. Nevertheless, judgement of its benefit depends on subjective preferences. Supposedly the impacts are viable, it is likely the precipitation, temperature, view distance, wind speed and rain and bad weather identification that has the largest impact.

In reference to the two preceding questions, the best performing feature set in our research consists of the basic sub-set, the locations sub-set version 1 and train interactions sub-set version 3 (B0L1I3). That implies the feature set is composed of the following individual features:

- *Basic sub-set*: Day of the week, Hour, Minute, Location, Direction, Current delay, Delay one and two registration points before
- *Locations sub-set* version 1: Total frequency of departures, short departures and pass-through points
- *Interactions sub-set* version 3: Expected headway to the first IC, SPR and LM train

In addition to that, some features from the other categories have a potential in delay prediction and those are certain weather conditions features, especially precipitation and temperature, possibly also view distance, wind speed and rain or bad weather identification.

Compared to the Naïve forecast, the feature set B0L1I3 (see the preceding questions) brings an improvement in the RMSE of 0.51, thus the RMSE changes from 1.99 to 1.48, or likewise the MSE changes from 3.94 to 2.20. The MSE 95% confidence interval shifts from [3.90, 3.98] to [2.17, 2.23]. The feature set reaches the lowest RMSE by 29% of train series which is the highest share among the feature sets. With respect to delay existence prediction, the F1 score of the Naïve forecast is 0.586 while the selected feature set reaches a score of 0.665. Also the precision score shows an improvement (from 0.462 to 0.559) and so does the recall (from 0.798 to 0.820) and accuracy (from 0.898 to 0.912). Delay change and delay jump prediction was not evaluated for the Naïve forecast due to its nature of rigorously 'predicting' delay decrease by 1 minute. Nevertheless, the F1 scores of the decreasing and increasing classes of delay change prediction by the finale feature set are 0.451 and 0.245 respectively. Recall scores of the classes are 0.332 and 0.155 respectively and precision scores 0.701 and 0.579 in the corresponding order. The F1 score of delay jump prediction then is 0.090 with precision of 0.050 and recall 0.470.

VI. Conclusions

Compared to the Naïve forecast, the feature set B0L1I3 brings an improvement in the RMSE of 0.51, thus the RMSE changes from 1.99 to 1.48, or likewise the MSE changes from 3.94 to 2.20. The MSE 95% confidence interval shifts from [3.90, 3.98] to [2.17, 2.23]. The feature set reaches the lowest RMSE by 29% of train series, which is the highest share among the feature sets. With respect to delay existence prediction, the F1 score of the Naïve forecast is 0.586, while the selected feature set reaches a score of 0.665. Also the precision score shows an improvement (from 0.462 to 0.559) and so does the recall (from 0.798 to 0.820) and accuracy (from 0.898 to 0.912). Delay change and delay jump prediction was not evaluated for the Naïve forecast due to its nature of rigorously 'predicting' delay decrease by 1 minute. Nevertheless, the F1 scores of the decreasing and increasing classes of delay change prediction by the finale feature set are 0.451 and 0.245, respectively. Recall scores of the classes are 0.332 and 0.155, respectively, and precision scores 0.701 and 0.579 in the corresponding order. The F1 score of delay jump prediction then is 0.090 with precision of 0.050 and recall 0.470.

The best performing model was not the one built on the most complex feature set consisting of features from all the categories. The most complex feature set (B0L1P0W0I3 consisting of feature sub-sets base, locations v.1, passengers v.0, weather v.0 and interactions v.3) was burdened by a RMSE 1.50 (correspondingly by a MSE 2.25 with a 95% confidence interval [2.21, 2.28]) and RWMSE 1.36, while the finally selected feature set (B0L1I3 consisting of feature sub-sets base, locations v.1and interactions v.3) reached RMSE 1.48 (correspondingly a MSE 2.20 with a 95% confidence interval [2.17, 2.23]) and RWMSE 1.35. Although the B0L1P0W0I3 feature set outperformed most of the other feature sets in delay change prediction, it lagged behind with respect to all the other performance indicators. According to the results of models consisting passengers features and the feature importance analysis, it is the passengers features that possibly worsen the prediction performance. On the other hand, the train interactions features and some of the weather features do contribute to the prediction quality according to the applied performance indicators. The train interactions features have a strong improving effect on the performance with respect to all the performance indicators, while weather features affect especially the resulting delay change prediction

References

Albert, S., Kraus, P., Müller, J. P., & Schöbel, A. (2017). Passenger-Induced Delay Propagation: Agent-Based Simulation of Passengers in Rail Networks. *Simulation Science*, 3–23. https://doi.org/10.1525/california/9780520292765.003.0006

Berger, A., Gebhardt, A., Müller-Hannemann, M., & Ostrowski, M. (2011). Stochastic Delay Prediction in Large Train Networks. In *11th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems* (pp. 100–111). https://doi.org/10.4230/OASIcs.ATMOS.2011.100

Chen, M., Liu, X., Xia, J., & Chien, S. I. (2004). A dynamic bus-arrival time prediction model based on APC data. *Computer-Aided Civil and Infrastructure Engineering*, *19*(5), 364–376. https://doi.org/10.1111/j.1467-8667.2004.00363.x

Chen, M., Yaw, J., Chien, S. I., & Liu, X. (2007). Using automatic passenger counter data in bus arrival time prediction. *Journal of Advanced Transportation*, *41*(3), 267–283. https://doi.org/10.1002/atr.5670410304

Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 785–794). San Francisco. https://doi.org/10.1145/2939672.2939785

Corman, F., & Kecman, P. (2018). Stochastic prediction of train delays in real-time using Bayesian networks. *Transportation Research Part C: Emerging Technologies*, *95*, 599–615. https://doi.org/10.1016/j.trc.2018.08.003

Cox, T., Houdmont, J., & Griffiths, A. (2006). Rail passenger crowding, stress, health and safety in Britain. *Transportation Research Part A: Policy and Practice*, *40*(3), 244–258. https://doi.org/10.1016/j.tra.2005.07.001

Goverde, R. M. P. (2010). A delay propagation algorithm for large-scale railway traffic networks. *Transportation Research Part C: Emerging Technologies*, *18*(3), 269–287. https://doi.org/10.1016/j.trc.2010.01.002

Hansen, I. A., Goverde, R. M. P., & Van Der Meer, D. J. (2010). Online train delay recognition and running time prediction. In *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*. https://doi.org/10.1109/ITSC.2010.5625081

Huisman, T., & Boucherie, R. J. (2001). Running times on railway sections with heterogeneous train traffic. *Transportation Research Part B: Methodological*, *35*(3), 271–292. https://doi.org/10.1016/S0191-2615(99)00051-X

INFORMS. (2018). RAS Problem Solving Competition. Retrieved December 18, 2018, from https://connect.informs.org/railway-applications/awards/problem-solving-competition

Kecman, P. (2014). *Models for Predictive Railway Traffic Management*. Delft University of Technology, Delft. https://doi.org/10.4233/uuid:539e96b3-18d7-4f6f-b662-ce22ae269f2a

Kecman, P., & Goverde, R. M. P. (2015a). Online Data-Driven Adaptive Prediction of Train Event Times. *IEEE Transactions on Intelligent Transportation Systems*, *16*(1), 465–474. https://doi.org/10.1109/TITS.2014.2347136

Kecman, P., & Goverde, R. M. P. (2015b). Predictive modelling of running and dwell times in railway traffic. *Public Transport*, *7*(3), 295–319. https://doi.org/10.1007/s12469-015-0106-7

Lessan, J., Fu, L., & Wen, C. (2019). A hybrid Bayesian network model for predicting delays in train operations. *Computers and Industrial Engineering*, *127*, 1214–1222. https://doi.org/10.1016/j.cie.2018.03.017

Ling, X., Peng, Y., Sun, S., Li, P., & Wang, P. (2018). Uncovering correlation between train delay and train exposure to bad weather. *Physica A: Statistical Mechanics and Its Applications*, *512*, 1152–1159. https://doi.org/10.1016/j.physa.2018.07.057

Nabian, M. A., Alemazkoor, N., & Meidani, H. (2018). *Predicting Near-Term Train Schedule Performance and Delay Using Bi-Level Random Forests*.

Nair, R., Hoang, T. L., Laumanns, M., Chen, B., Cogill, R., Szabó, J., & Walter, T. (2019). An ensemble prediction model for train delays. *Transportation Research Part C: Emerging Technologies*, *104*(April), 196–209. https://doi.org/10.1016/j.trc.2019.04.026

Oneto, L., Fumeo, E., Clerico, G., Canepa, R., Papa, F., Dambra, C., … Anguita, D. (2016). Advanced analytics for train delay prediction systems by including exogenous weather data. In *IEEE International Conference on Data Science and Advanced Analytics* (pp. 458–467). https://doi.org/10.1109/DSAA.2016.57

Peters, J., Emig, B., Jung, M., & Schmidt, S. (2006). Prediction of Delays in Public Transportation using Neural Networks, 92–97. https://doi.org/10.1109/cimca.2005.1631451

San, H. P., & Mohd Masirin, M. I. (2016). Train dwell time models for rail passenger service. *MATEC Web of Conferences*, *47*. https://doi.org/10.1051/matecconf/20164703005

Tirachini, A., Hensher, D. A., & Rose, J. M. (2013). Crowding in public transport systems: Effects on users, operation and implications for the estimation of demand. *Transportation Research Part A: Policy and Practice*, *53*, 36–52. https://doi.org/10.1016/j.tra.2013.06.005

Van den Bulk, L., Fioole, P., & Kachergis, G. (2018). *Predicting Short Term Train Delays in the Dutch Rail Network*. Utrecht.

Wang, R., & Work, D. B. (2015). Data Driven Approaches for Passenger Train Delay Estimation. In *2015 IEEE 18th International Conference on Intelligent Transportation Systems*. Las Palmas, Spain. https://doi.org/10.1109/ITSC.2015.94

Yaghini, M., Khoshraftar, M., & Seyedabadi, M. (2013). Railway passenger train delay prediction via neural network model. *Journal of Advanced Transportation*, *47*, 355–368. https://doi.org/10.1002/atr

Yuan, J. (2007). Dealing with Stochastic Dependence in the Modeling of Train Delays and Delay Propagation. In *International Conference on Transportation Engineering 2007* (pp. 3908–3914). Reston, VA: American Society of Civil Engineers. https://doi.org/10.1061/40932(246)641

Yuan, J., Goverde, R. M. P., & Hansen, I. a. (2002). Propagation of train delays in stations. *Computers in Railways Viii*, *13*, 975–984\n1163.

Table of content

List of tables

List of figures

1. Introduction

Railways have always played an important role in passenger transport and it becomes increasingly important as there is a world-wide need for changes in transportation to address increasing demand for mobility, together with increasing population, while decreasing emissions generated by the transport sector. Extensive research is being performed to provide high-quality services with low negative externalities. Public transport, and thus also railway transport, certainly is a crucial element in the future of passenger transport. Yet, there always will be competition with other modes and therefore, the strive to provide better service than yesterday is unending. Besides making railways attractive to users of other modes, comfort and satisfaction of the current users are a priority as well. Achieving these goals does not consist only of large easily visible actions. There are many incremental innovations that bring seemingly small improvements which together can make a big difference. An example of such an additional element in the service quality is an improved passenger information about train delays.

Understandably, prevention of delays or an appropriate response to emerging delays should be a primary concern. Our research does not aspire to prevent delays, instead, our work aims to forecast the delays' length in the near future, possibly in the whole corresponding railway network. Prior knowledge about upcoming delays may mitigate inconvenience induced to the passengers by providing them with an opportunity to change their itineraries, or possibly might even eventually help to prevent delay propagation by prompt dispatching interventions.

Our research has been conducted in cooperation with Nederlandse Spoorwegen, commonly referred to as NS, the principal passenger railway operator in the Netherlands who provided us with data, resources and valuable knowledge to carry out our study, and to test and assess our models on the Dutch passenger railway network. Besides NS, valued support was accommodated also by ProRail, a rail infrastructure manager in the Netherlands, who contributed by providing a substantial part of the data we used in our work.

In the Introduction chapter, we will provide a deeper explanation of the delay prediction problem. Then, a brief overview of the current state of the research in this field is presented and the resulting research gap is deduced. Consequently, the research objective and the research question and sub-questions are defined, and finally, an overview of our research approach is given.

1.1. Problem statement

Aiming for the highest service quality possible, NS seeks a constant improvement focusing on its performance and satisfaction of its customers. The latter subject of focus, customer satisfaction, was also defined as one of the key objectives for 2019. A yearly set of performance indicators contains an indicator defined as '*Information on the train and at stations about delays*'. Although the target value for the year 2019 was set to 80%, this threshold already had been reached in the previous years as it was 83.2% in the year 2017 and in the year 2018, the measured value further increased by 2%-points and reached 85.2%. (NV Nederlandse Spoorwegen, 2019) Information whether and how much a train is delayed is displayed at the stations, and online platforms such as NS website or travel planners. The NS planner application considers expected delays in the search for possible connections and disregards those with an impossible transfer due to the expected delays. The accuracy of current train departure delay prediction 20 minutes to the future was observed by Fioole (2018b) who registered all estimated and realized passenger train departures from all stations in the Netherlands over two days in November 2018, three days in December and three days in January 2019. In total, 78 362 train departures were registered. The observed data showed that between 94.7 % and 98% of the predicted departures were forecasted correctly as either on time or late, where '*on time*' included departures with up to 2 minutes delay, and '*late departures*' were delayed by more than 2 minutes. The 2 minutes threshold was based on a threshold that was used for punctuality measurements at NS in the past though it was replaced by a threshold of 5 minutes of a passenger delay recently, which however did not correspond well with the purpose of the work of Fioole.

Although the accuracy may seem to be relatively high, it is necessary to consider that the vast majority of the departures was on time. In an extreme case, predicting all departures to be on time would result in a comparably high accuracy simply because of the disproportion in observed on-time and late departures. The accuracy of predicting actually delayed departures was significantly lower as realized late train departures were predicted to be late in only approximately 34%-40%. In the 8 measurement days, approximately 1900 delayed departures were wrongly predicted as on time while eventually being late. Due to the nature of the currently applied delay prediction, delays tend to be underestimated and it is far more common that a train departure is wrongly expected to be on time than falsely forecasted as delayed.

In the current delay forecast, two methods are available at NS and ProRail, '*Naïve forecast*' and '*Train delay forecast*' by Brouwer (2012). The latter method was proposed by an NS intern in the year 2012 who used historical data of individual trains to predict delays. The method performed well in simple predictions however could not predict large changes in delays with a sufficient reliability (INFORMS, 2018d) and thus is not currently applied. The first method, '*Naïve forecast*', is therefore of higher importance in our research, as its performance is the main benchmark used for assessment of our models' performance.

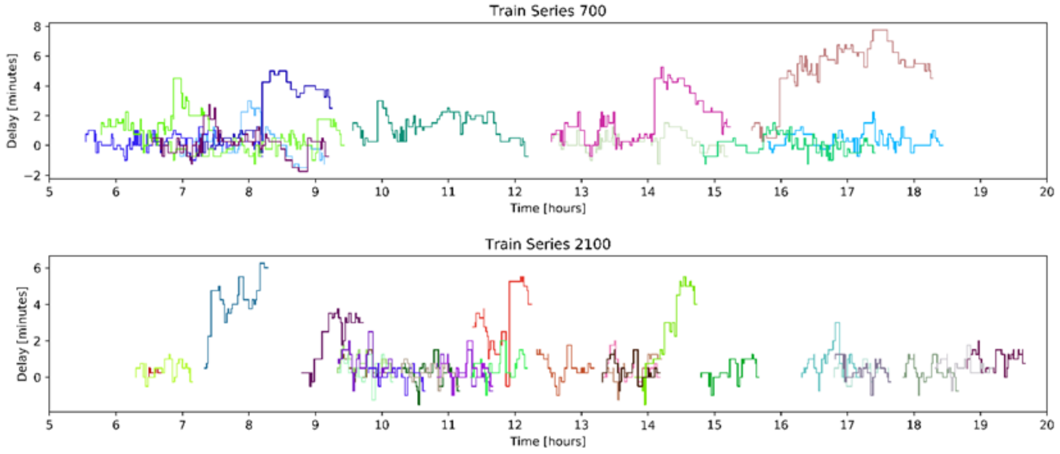


Figure 1 Delay development over time of a random sample of trips by two randomly chosen train series.

The Naïve forecast method assumes a delay is always reduced by the slack time which is 5% of the bare driving time. In the span of 20 minutes, any delay is therefore assumed to get reduced by 1 minute. Important to say, this assumption is applied for passenger information only. In contrary, for dispatching purposes, delays are assumed to remain unchanged from the current state. Understandably, the major inaccuracy of the Naïve forecast method lies in the assumption that a delay always gets reduced over time which often is not the case as practice reveals. An example of this fault is illustrated in Figure 1 where a delay development along a set of randomly selected trips of trains belonging to two train series is depicted, and where one can observe a fluctuation of the delays in both directions up and down with a varying gradient of the change.

1.1.1. RAS Problem Solving Competition 2018

A convenient way to explore numerous innovative methods, in addition reflecting the local specifics, in this case characteristics of the Dutch railway, was to reach out to students and experts in the field all around the world via an international competition. NS in cooperation with ProRail therefore took part in INFORMS Railway Problem Solving Competition (INFORMS, 2019b) and prepared an assignment with the title '*Predicting Near Term Train Schedule Performance and Delay*' which was announced in spring 2018. Over 40 groups participated in the competition and proposed various methods to tackle the problem. A committee consisting of experts from NS and ProRail selected the best submissions based on accuracy of the solution, sophistication and robustness of the method and quality of the submitted paper (INFORMS, 2018b). The winners were announced at the INFORMS conference in Phoenix, USA in the autumn of 2018 (INFORMS, 2018e).

The assignment was publicly available on the website of INFORMS (2018c) along with relevant data and the necessary instructions. Being it an annual competition, all the documents were moved to an archive page of the website and can no longer be accessed via the original links. Instead, an interested reader may refer to the archive available on the website of INFORMS (2019).

The task was to develop an approach to predict delays of passenger trains operated by NS 20 minutes to the future based on knowledge of the present state of the network. As a case, historical data were provided by NS and ProRail. The data referred to a period between September and December 2017 and consisted of the planned timetable, corresponding realization data revealing delays of the trains at relevant registration points, crew schedules, rolling stock circulation plan, basic infrastructure description and simple weather conditions information (INFORMS, 2018c). A thorough description of part of the data can be found in Chapter 4 as some of the datasets were used also in our research.

Train delays were supposed to be predicted in three categories of delay development: a *'change'*, i.e. prediction of the direction of a delay change as decrease or increase if such change is at least 2 minutes or constant otherwise, a *'jump'*, i.e. prediction of a change in delay by 4 or more minutes in either direction, and *precise prediction* of the actual delay in minutes. The categories were designed to address different levels of complexity of delay prediction: reaching higher accuracy in forecasting solely whether a delay will increase, or decrease was considered to be easier than forecasting large changes in delay and even easier than predicting actual magnitude of the delays. Besides the varying difficulty of the tasks, they were intended to tackle challenges of predicting large delay changes and correct prediction of the delay change direction.

In the meantime, an intern with background in mathematics was hired at NS and was given the same task: the same assignment with the same data available. Her research (Van den Bulk, Fioole, & Kachergis, 2018) served as a reference for assessment of the submissions by the competition participants. Interestingly, methods proposed by the two winning teams matched methods proposed by Van den Bulk et al. who recommended the use of machine learning techniques, particularly a decision trees based gradient boosting method called XGBoost developed by Chen & Guestrin (2016) and Neural networks. Neural Networks was a method proposed by the team of Hellsten, Haahr, & van der Hurk, (2018), while the second winning team of Nabian, Alemazkoor, & Meidani (2018) suggested using Bi-level random forests, a decision tree based method.

As the performance of the method was not the only assessment criterium in the competition, the final position of the teams does not directly indicate the best method for the problem. Furthermore, the models proposed by the two winning teams performed comparably and it therefore cannot be concluded which method suits the problem better solely based on the competition results. Above that, Van den Bulk et al. (2018) came to a similar conclusion that performance of Neural Network and decision trees based method, XGBoost in their case, was comparable applied on the described problem.

Only summaries of results of the models by Hellsten et al. (2018), Nabian et al. (2018) and Van den Bulk et al. (2018) were available providing merely a brisk insight into their models' performance. The models proposed by Hellsten et al. (2018), Nabian et al. (2018) correctly predicted a train running on time (including a delay up to 2 minutes) or being late in 95.96% and 91.94% respectively. However, false delay forecast or false prediction of a train running on time is of higher importance. Trains having a delay in the future state were correctly predicted to be delayed only in 18.75% of cases by Hellsten et al. (2018) and in 25.69% cases by Nabian et al. (2018), the rest of delayed trains were forecasted to run on time. The currently used model correctly predicts delays in approximately 34%-40% (Fioole, 2018b). Inversely, trains predicted to have a delay actually were on time in 18.18% of the cases a delay was predicted according to the model by Hellsten et al. (2018) and in 78.61% of cases according to the model by Nabian et al. (2018). Currently, false predictions of delays occur in about 9%-10% of the cases a delay was predicted. It is thus obvious that the model by Nabian et al. (2018) tends to significantly overestimate delays. An overview of the mentioned examples is in Table 1. It is worth mentioning that performance of both models by Hellsten et al. (2018) and Nabian et al. (2018) in the comparison with the current state is biased by a narrow testing instance which moreover consists in two thirds of time during peak hours. Furthermore, the context in which the models were developed, that is for a competition, suggests the models might not be deployed to their fullest potential.

Results given by Van den Bulk et al. (2018) are in a format which is not comparable with above discussed results. Their own comparison to the currently applied method however shows a significant improvement by applying XGBoost model and Neural network model with a variety of features considered. For example, the F1-score, i.e. a harmonic mean of ratios of falsely assigned class and falsely not assigned class (Sokolova & Lapalme, 2009), of a model assuming a delay does not change is 0.30 for the label *'change'*, i.e. prediction of a direction of delay change, while XGBoost model reaches up to 0.50 and Neural network model 0.33. Further, Van den Bulk et al. (2018) score performance of the prediction of the actual delay by root mean squared error (RMSE). Note that they used RMSE for its interpretability as it represents an average error of the prediction in minutes. The model assuming a constant delay is burdened by RMSE of 1.53, while XGBoost application of Van den Bulk et al. reaches RMSE of 1.37 or 1.34 for two different feature sets. Neural network models by Van den Bulk et al. then reach RMSE of 1.38 or 1.35 for two feature sets. An overview of the values is in the

Table 2.

Table 1 An overview of examples of delay prediction accuracy of different models

	Hellsten et al. (2018)	Nabian et al. (2018)	Current model (Fioole, 2018)
Correct prediction (delayed or on time)	95.96 %	91.94 %	94 % - 98 %
Correct prediction (delayed)	18.75 %	25.69 %	34 % -40 %
False prediction of a delay	18.18 %	78.61 %	9 % - 10 %

Table 2 Example of comparison of methods' accuracy according to Van den Bulk et al. (2018).

	No change in delay	XGBoost	Neural Network
F1-score: 'change' label	0.30	0.50	0.33
F1-score: 'jump' label	-	-	0.12
RMSE: actual delay prediction	1.53	1.37 / 1.34	1.37 / 1.34

The models developed by Van den Bulk et al. (2018) thus show a certain improvement in comparison to the delays being assumed to remain unchanged in the terms of prediction accuracy. The results therefore support investing further research efforts in this direction.

1.1.2. What next?

Logically, the models developed for the RAS competition were built solely on data and information provided in the competition which were highly limited for understandable practical and legal reasons. Many factors that might be influencing the delays development were therefore omitted and left space for possible improvements and further research. Based on interviews at NS, there are two main domains that are expected to have a significant effect on delay development: interactions among trains sharing the infrastructure, specifically represented by headways (Fioole & Tielman, 2019), and effects of passenger crowding (Fioole, 2018a). Nonetheless, it is not a complete list of all potential factors that may be used for delay prediction. Many delay sources are mentioned in the literature, frequently as an incomplete list of examples. Some of the examples complemented by highly detailed delay source categorization by Union Internationale des Chemins de Fer (UIC; International Union of Railways) are presented in Section 2.2. As a concise introduction to the problem, a handy comprehensive categorization of delays according to what part of the railway system the delays relate to was adapted from work of Lee, Yen, & Chou (2016). Within each of the categories, they also provide an illustrative list of possible delay causes giving us a handy overview of factors possibly relevant for our research of a delay forecast (see Table 3).

Table 3 Possible sources of delays in passenger railways (Lee, Yen, & Chou, 2016)

Category	Potential delay source
Station	· Dispatching · Service preparation · Boarding/alighting of passengers
Train	· Rolling stock failure · Weather conditions · Accidents, Facility failure
Operations	· Signaling system failure · Track or facilities maintenance
Timetable	· Slack time availability · Estimated bare driving time · Track configuration · Rolling stock characteristics · Overtaking and Meeting, Track assignment

Important criteria for the purpose of our work are that the delay sources (delay development indicators) must be observable and/or applicable in real time, quantifiable and fully recorded to allow real-time application of a machine learning prediction model. Dispatching interventions are hardly predictable as protocols for specified disruptions serve only as a guideline and the final interventions depend on an individual's expertise. Moreover, dispatching interventions rather introduce noise in the observed data as they are not fully recorded and cause inconsistency in train numbering or in the scheduled times leading to falsely identified delays (as for example a train appears to be delayed by a significant amount of time while in reality, the train was swapped with another one, running relatively according to the schedule for the passengers, only with a different train number registered in the data). Next, delays in the service preparation processes at the stations are likely to be difficult to enumerate and register.

Unusually high numbers of passengers on the platforms or in the vehicles may lead to extended dwell times (Lam, Cheung, & Lam, 1999). The exact numbers of passengers that are going to board or leave a train are unknown for the future locations. However, based on fare collection data, it is possible to estimate the numbers of passengers that boarded or left the train at the past stations which might indicate uncommon states of the system with respect to the passengers volume.

Rolling stock failure, accidents, facility failure or signaling system failures are all events that often lead to extensive disruptions and the effects of such events are a large research topic on its own. Importantly, any of these events necessitate a significant dispatching intervention. As explained earlier, dispatching decisions are hardly predictable, however above that, the need for interventions also means that there is a lack of time and attention that can be paid to register such an event in a way that could be directly processed in a delay forecasting model especially on a macroscopic level. Contrary to failures in the railway system, maintenance of the facilities and especially of the track can cause significant delays. Though it is generally a planned action and thus is known in advance, its quantification and translation into factors usable in our research is questionable.

Finally, timetable-based sources of delays are relatively easily obtainable. It can be assumed that in the case of the railway system in the Netherlands, the slack time is fixed to 5% as mentioned earlier. Rolling stock characteristics are determined by the train number or the train series, track configuration by the location, and so is the bare driving time. Eventually, interactions among trains due to the track assignment and availability of passing points are retrievable from the planned track assignment. Although the assigned track may differ in reality due to dispatching interventions, such changes are rather difficult to register in real time.

As was mentioned earlier, the factors suitable for application in delay prediction in the context of our work should be observable in the whole network in an automated way, should be quantifiable and should comprise all relevant occurrences. Based on the discussion above, that can be achieved in the case of using *passengers*, *weather conditions* and *timetable related delay sources*. The most important features identified by Van den Bulk et al. (2018) were the *current hour*, *minutes* within that hour and the *location*. It was assumed that it might have to do for example with rush hours and sizes of stations. It therefore supports the suggestion made by Fioole (2018a) that it may be beneficial to include crowding at the platforms and/or on the trains in the model.

Although a dataset containing information about weather conditions in the Netherlands was provided along with the other data available in the RAS competition (INFORMS, 2018a), the models by Hellsten et al. (2018) and Van den Bulk et al. (2018) did not contain any features reflecting weather conditions. Nabian et al. (2018) did include features reflecting the weather conditions, however, the features were discarded due to low feature importance score. Likely, that was because of the data's low precision as weather conditions were described by daily averages of the country as a whole.

Van den Bulk et al. (2018) attempted incorporating train interactions in one of the feature sets, defining the interactions as "*if another train series passes through a location that the current train series also passes through*" (Van den Bulk et al., 2018, p.7). The value of the feature was equal to delay of the train passing through the nearest shared location before the investigated train. If no location was shared with another train within 20 minutes, the feature was set to zero. A set of features associated with each train series would therefore contain an array of interactions with all other train series in both directions. Implementation of these features in XGBoost, however, did not show any significant improvement in comparison to other feature sets, and even lead to a decline in accuracy. That is explained by the authors as that XGBoost uses only a subset of features per tree, and the extensive number of features then causes inefficient branching. On the other hand, an application in Neural Networks showed slight improvement in predicting the direction of a delay change. Nevertheless, both models performed significantly worse in estimation of actual magnitude of the future delay. That is explained by the authors as that the exact delay of the interacting train does not directly affect delay of the succeeding train. Replacement of the exact delay in the feature set by binary coding, representing only existing interacting trains that have any delay, or any delay above a specified limit, is suggested. Nevertheless, the data regarding train interactions available for that study were limited and detection of train interactions were therefore imprecise. Most importantly, sharing a location, such as a station, does not necessarily mean also sharing a piece of track and therefore there might not be any interaction in reality. Furthermore, different definition of the features also may have an effect on the impact of their implementation in the models. Potential influence among the trains on delays of others is therefore hardly retrievable from the original dataset. Hellsten et al. (2018) incorporated in their model an indicator of track segments defined as a connection between two consecutive events. Interpretation of the segments usage in the terms of the traffic density or train interactions was left upon the model. The influence of specifically those features unfortunately is not elaborated on. Nabian et al. (2018) defined a set of features reflecting a frequency of the historical events with leading train delays where the leading trains were identified as the trains that share at least one location with the train subject to the prediction within the schedule. The features however were neglected due to their low scoring in feature importance assessment.

Research gap

A brief insight into the available scientific literature along with the above introduced work of Hellsten et al. (2018), Nabian et al. (2018) and Van den Bulk et al. (2018) support or do not oppose an inclusion of passenger counts, weather conditions and train interactions as features in to the selected model (XGBoost) for train delay prediction. Note that a more detailed overview of the literature is presented in Chapter 2.

The most commonly used factors in delay prediction in railways are network- and timetable-based such as departure and arrival times, location identification, event type, train identification and similar (Hansen, Goverde, & Van Der Meer, 2010; Kecman, 2014; Kecman & Goverde, 2015a; Lessan, Fu, & Wen, 2019; Nair et al., 2019; Oneto et al., 2016; Wang & Work, 2015; Yaghini, Khoshraftar, & Seyedabadi, 2013). Similar attention has been paid to delay propagation and its prediction (Berger, Gebhardt, Müller-Hannemann, & Ostrowski, 2011; Corman & Kecman, 2018; Goverde, 2010; Peters, Emig, Jung, & Schmidt, 2006; Jianxin Yuan, 2007). However, in contrary to the lengthy list of publications considering the earlier factors in train delay prediction models, weather conditions were considered in much fewer models (Nair et al., 2019; Oneto et al., 2016) with a support of an observed correlation between weather conditions and train delays (Ling, Peng, Sun, Li, & Wang, 2018). Passenger counts and the effects of crowding attract a lot of attention among transport researchers. In general, especially dwell time (Kecman & Goverde, 2015b; San & Mohd Masirin, 2016) and psychological effects of crowding (T. Cox, Houdmont, & Griffiths, 2006; Tirachini, Hensher, & Rose, 2013) are in the spotlight. However, application of the passenger counts and crowding in the railway delay prediction is rather scarce (Albert, Kraus, Müller, & Schöbel, 2017), though some examples can be found in the bus transit domain (M. Chen, Liu, Xia, & Chien, 2004; M. Chen, Yaw, Chien, & Liu, 2007).

All the selected factors, passenger counts, weather and train interactions (in the meaning of delay propagation) were somehow implemented in some delay prediction model. However, none of the publications reviewed presented an attempt to include all the factors into one model. Furthermore, many types of models were presented in the reviewed publications such as stochastic models (Berger et al., 2011; Huisman & Boucherie, 2001; Yuan, Goverde, & Hansen, 2002), Bayesian networks (Corman & Kecman, 2018; Lessan et al., 2019), data mining techniques (Hansen et al., 2010) or Neural network (Peters et al., 2006; Yaghini et al., 2013; Oneto et al., 2016). Decision trees based models appeared mainly in the form of a random forest model (Oneto et al., 2016; Nair et al., 2019). No application of gradient boosting model, such as XGBoost, was found in the context of train delay prediction, and especially not in a combination with all the factors affecting delays as mentioned above. Finally, most of the publications that were offering any comparison of multiple models focused on assessment of the differences in the models' architecture rather than comparing an impact of various factors considered in the models.

The consequent research gap was found in the following three aspects:

- Application of a gradient boosting decision trees-based model, specifically XGBoost.
- Inclusion of passenger counts data into a passenger trains delay prediction model.
- Combination of timetable-based information, weather condition, passenger counts and train interactions for delay prediction, implementing each factor separately and in various combinations with the other factors.

1.2. Research Objective and Research Question

The objective of our research is to define and evaluate sound feature sets reflecting effects of passenger counts, weather conditions and train interactions on passenger trains delays, which are to be used along with features derived from data available in the RAS competition to forecast the passenger trains' delays 20 minutes to the future. Effects of the individual features and their combinations are to be reflected upon. Note that the span of 20 minutes to the future refers to the last registration point within a 20-minutes time window since the last registration point.

The model will be based on the conclusions drawn by Van den Bulk et al. (2018). Specifically, a decision trees based gradient boosting method XGBoost will be used. The research will be carried out in the context of the Dutch railway system, using relevant historical data. Performance of the forecast will be measured by defined key performance indicators and in context with the prediction categories defined in the RAS competition. Criterion for performance measure is accuracy of delay predictions compared to the currently applied 'Naïve forecast' model and secondarily to work of Van den Bulk et al. (2018).

Research question

Resulting from the research objective, the research question is defined as follows.

> What is the performance of passenger trains' delays forecast 20 minutes to the future if a machine learning model, built on previous research by NS and extended by passenger and weather data and offline identified and online updated train interactions, is used, and how do these factors contribute?

Sub-questions

To set a direction to reach the answer to the main research question, a set of sub-questions was defined. These questions provide a guidance through the main steps of our research: theory understanding, system analysis and data description, model development, and finally the results evaluation. Answers to these questions will close and summarize relevant chapters throughout the report.

Table 4 Research sub-questions

Theory	1	Why and how should the defined factors be included in the delay prediction model?
	2	What are appropriate key performance indicators considering the model's characteristics?
System analysis & data	3	How can/must the data representing the factors be modified to be used as features in the models expectedly maximizing their contribution to the model's performance?
	4	Is there an observable relationship between the included factors and train delays?
Modelling	5	What are sound feature sets?
	6	What is a sound model structure including parameter tuning?
Evaluation	7	What is the performance of the feature sets relatively to each other and to the reference model?
	8	What is the importance of the features within the feature sets?
	9	What features/feature sets have the highest/lowest potential in delay prediction?
	10	What is the forecast potential of the highest performing feature set(s)?

1.3. Research approach and document structure

The document is structured in line with the research approach. *Chapter 2 Literature research* provides a broader theoretical background necessary for better understanding. That includes a review of literature concerning delays in passenger railways along with an overview of the recent research in the field of delay prediction in passenger railways. Next, the effects of passenger crowding and weather conditions, complemented by the problem of delay propagation due to train interactions are presented. Finally the first sub-question is answered.

The following *Chapter 3 Method* then continues in a theoretical manner, providing necessary understanding of the method applied. Furthermore, key performance indicators are defined in accordance with the method applied, answering the sub-question 2.

Chapter 4 Data gives a detailed description of the data used in our research. The sources of the data are mentioned, so is the original data structure and what features are retrieved from the data. Description of the data processing that was carried out to clean the data and to turn them into a convenient form is available in the Appendix A. At the end of Chapter 4, a brief data analysis is presented, and finally, a summary and answers to the sub-questions 3 and 4 are given.

In the following step, in *Chapter 5 Model development and tuning*, the complete feature sets are introduced along with a scheme to test and compare prediction performance of their various combinations. Furthermore, the design of the model's structure is presented. That includes the model's configuration and an algorithm of parameters tuning. In the summary of the chapter, the sub-questions 5 and 6 are answered.

In *Chapter 6 Research results*, we provide the consequent results. The presentation of results is structured according to the various key performance indicators defined in Section 3.5.1. Results of the numerous feature sets are compared, and the best performing feature set is selected and analyzed in more detail. The results provide answers to the sub-questions 7 to 10.

Finally, discussion of the results and a reflection of the research in a more critical and broader point of view is provided in *Chapter 7 Discussion* followed by the final *Chapter 8 Conclusion*. The research is summarized, main findings are mentioned, and final conclusions are drawn. Eventually, recommendations for further research are proposed and the practical implications of our research outlined.

2. Literature research

In the previous chapter, we explained the motivation for our research in a practical context (as a continuation of the previous research initiated by NS) and by discovering a scientific research gap. That resulted in the definition of a research question and the auxiliary sub-questions. In the literature research, we want to expand the motivation for the inclusion of the selected factors and to deepen the understanding of the delay prediction problem in a broader context. We begin by clarifying the term '*delay'* along with its role in the passenger railway system. Next, we explore where the delays may originate from in the system. That is followed by an overview of the methods that have been applied to the delay prediction problem in railways, from which we seek an inspiration for our work with respect to the method we apply. Finally, we zoom in on the factors that we decided to include in our delay prediction model to research their role and behavior in the railway system.

2.1. Delays in passenger railway

The Cambridge Dictionary defines the word 'delay' either as a verb: "*to make something happen at a later time than originally planned or expected*" or "*to cause someone or something to be slow or late*", or as a noun: "*the situation in which you have to wait longer than expected for something to happen, or the time that you have to wait*" (Cambridge University Press, 2019a). Translation of the first two definitions to the railway context could be written as 'a train activity taking place at a later time than originally planned or expected' and 'causing passengers to be slow or late'. The second definition is clear and closely relates to the definition of delay as a noun. However, the first definition provides space for interpretation of a delay thanks to the formulation: "*at a later time than originally planned or expected.*".

When assessing service qualities based on delays, the relevant measure is punctuality. Its definition being "*the fact of arriving, ... at the expected or correct time and not late* " (Cambridge University Press, 2019b) is obviously largely similar to the definition of a delay. The next related measure is reliability. Depending on the context, its definition and scope may vary. In the framework of our work, connectivity and travel time reliability as mentioned by Bell & Iida (1997) are the most relevant. Connected to the definition of reliability of a transport network as "*the probability that one or more of its links does not fail to function*" (Husdal, 2004, p. 188), we can understand reliability in the context of our work as the probability that passenger train connections do not fail to provide service according to their spatial and temporal planning. Hence, in sake of reliability, the train connections do not fail to ride without delays and changes in the routes. Finally, as reliability is one of the most important qualities in passenger railway operations and lies at the base layer of the Pyramid of customer needs together with safety (Van Hagen & De Bruyn, 2012), the importance of its maximization is apparent.

A question may be, whether delay prediction has the potential to address reliability. Supposedly a delay is known and communicated adequately in advance to accommodate necessary changes in personal or operational planning, the meaning of punctuality might slightly change reflecting on the word 'expected' in the definitions of punctuality and delay. Clearly, frequent or extensive delays would not lead to a reputation of a reliable service despite an accurate prediction and therefore there is a limit to which a delay prediction can improve passengers' experience.

First, criteria for delay acceptance can be the delay risk and the delay length, and its value in the eyes of passengers. This topic was the subject of work for example by Börjesson and Eliasson (2011) who conducted two stated choice surveys providing the participants with information about the expected travel time, travel cost, risk of delay and possible delay length. In the second survey, the variable of beforehand information about the length of a delay was added. Furthermore, the participants could choose a preferred improvement related either to the fare, delay length or delay risk, beforehand information or possible compensation. Results of the research showed that the value a delay does not increase linearly with the delay risk neither, the delay length. Instead, the increase of the value of a delay is rather slower and therefore, to some extent, the existence of a delay is essentially more distressing than its length. Unfortunately for our work, the authors did not elaborate further on the effect of the availability of a beforehand information about a delay, neither the preferred improvements.

Corresponding to availability of up to date information about delays, it is the unpredictability of public transit services that can be a significant source of stress and therefore can worsen its users' experience. Although public transit users reported lower stress levels due to delays than car users due to congestions according to a research by Gatersleben & Uzzell (2007), the perceived (un)predictability and resulting stress in a daily commute is a proven issue (Evans, Wener, & Phillips, 2002). The stress resulting from unpredictability of the services relates to a lack of control over the journey's characteristics (Sposato, Röderer, & Cervinka, 2012). And importantly, the longer the journey, the more unpredictable it is perceived to be in the eyes of the passengers (Gudden, 2014).

Finally, besides dissatisfaction or stress, delays cause loss to the passengers which may be monetarized. Using the Netherlands as an example, a governmental institute of the Netherlands named 'Kennisinstituut voor mobiliteitsbeleid' (translated as Knowledge institute for mobility) revealed in a publication 'Mobiliteitsbeeld 2017' (translated as Mobility Screening 2017) that approximately 685 million passenger-delay-minutes were accumulated by the NS passengers in the year 2016. That multiplied by an average value of time of 10.24 Euro/hour, as used in the publication, adds up to over 140 million Euro loss caused to NS passengers in the respective year (*Mobiliteitsbeeld 2017*, 2017). Value of time, and in the context of public transport, the value of travel time, is a fundamental research topic in the transport domain. Nonetheless, it is not only the total travel time that matters to the passengers, but importantly it is also the (un-)reliability of the expected travel time that comes with it. Bates, Polak, Jones, & Cook (2001) presented applications of utility theory proving the importance of public transport service punctuality and reliability to the passengers, which was in line with conclusions of Li, Hensher, & Rose (2010) who conducted a review of numerous studies relevant to passenger travel time reliability, considering car, bus and rail modes, and found that travel time reliability indeed is strongly significant in the decision-making process of passengers.

In summary, punctuality and resulting reliability is an important factor for users of public transit services. It is not the delay length that is the most important aspect, however, it is the delay presence and its risk itself. Furthermore, not only that delays cause disruptions in individuals' schedules and cause time and monetary loss. It is also an existence of the delay risk, which is a significant stress source, that is considered by the passengers while making decision what mode to take. Therefore, addressing the unpredictability of the service in terms of delay prediction might have a promising potential to increase comfort of the passengers in the meaning of decreased stress levels. Furthermore, prior knowledge of delays would give the opportunity to the passengers to adjust their schedules and travel plans and thus minimize their loss. Finally, the importance of developing an understanding of delays and improving their predictions is even larger if the operator's point of view is introduced. That is because there is a wide range of applications in railway operations such as real-time control for delay propagation reduction and connection management, passenger information, and estimation of transfers feasibility, and rolling stock and crew circulation optimization (Kecman, Corman, & Meng, 2015).

2.2. Sources of delays in passenger railway

Although delays may emerge for a large number of reasons, there is a major distinction that should be made before naming any. According to that distinction, delays are generally categorized as either primary or secondary, depending where they originated. Huisman & Boucherie (2001) define a primary delay as a delay that arises due to external factors while secondary delays are induced by delay propagation from other trains that have already been delayed. A more elaborate definition of a primary delay can be found in a Delay Attribution Guide issued by the Delay Attribution Board in the United Kingdom of Great Britain and Northern Ireland: "*A Primary Delay is a delay to a train that results from an Incident that directly delays the train concerned, irrespective of whether the train concerned was running to its schedule (schedule includes booked platform or line) at the time the incident occurred, i.e. the delay is not the result of another delay to the same or other train.*" (Delay Attribution Board, 2017, para. 2.7.2.). In some literature, the terms may be alternated by 'exogenous' and 'reactionary' delays (Gibson, Cooper, & Ball, 2002) or 'exogenous' and 'knock-on' delays (Carey, 1999) instead of primary and secondary delays, respectively. Similarly, the Delay Attribution Board refers to the secondary delays as 'reactionary' delays. Their definition of a reactionary (secondary) delay is: "*A Reactionary Delay is a delay to a train that results from an incident that indirectly delays the train concerned, i.e. the delay is the result of a prior delay to the same or any other train*" (Delay Attribution Board, 2017, para. 2.7.3.). Additionally, a distinction can be made between secondary delays caused by bunching, and by late arrival to a terminal station and consecutive propagation of a delay onto the following trip of the same vehicle. Murray & Grubesic (2007) further differentiates scheduled and unscheduled delays. Many other categorizations could be found, nevertheless, distinction of primary and secondary delays is the most important for our research.

Concerning the primary delay sources, Huisman, & Boucherie mention, for example, weather conditions or unusual passenger volumes, which can be complemented by examples given by Daamen et al. (2009) who point out technical failures on infrastructure or rolling stock. Mannhardt & Landmark (2019) further mention an important factor of human behavior and decision making in railway operations, such as decisions made by dispatchers or guards, or drivers' behavior. Although all literature sources, concerning problems related to delays in railway, name some delay sources, those mostly are only examples instead of complete and exhaustive lists. Naturally, one cannot anticipate all possible scenarios. Breaking down the railway system into functional segments and defining possible failures and events leading to delays within those segments is often used instead. An example has been provided in Section 1.1.2 in Table 3 where we presented delay categorization by Lee, Yen, & Chou (2016) who distinguishes delays in passenger railway as station-related (dispatching, service preparation, boarding and alighting of passengers), train-related (rolling stock failure, weather conditions, accidents, facility failure), operations-related (signaling system failure, track or facilities maintenance) and timetable-related (signaling system failure, track or facilities maintenance). Following categorization of factors used for delay prediction in a model developed by Nair et al. (2019), we can distinguish delay influencing factors as train-specific, infrastructure-related, network-related, train interactions and external factors.

Scientific articles mostly list only examples of possible delay causes in railway instead of structured and exhaustive enumerations of virtually all possible causes. For uniform delay source recording, International Union of Railways (UIC) provides a list of standardized delay sources which can be assumed to be highly exhaustive, despite that each category also includes a code for 'other source'. The list of categories and the enumerated causes defined by UIC were retrieved from a website of Telematics Applications for Passenger Services Technical Specifications for Interoperability (TAP-TSI, 2019). The categories and relevant delay sources are listed as follows:

Operational planning, Management

Timetable compilation; Formation of train if managed by Infrastructure Manager; Mistakes in Operational procedures; Wrong application of Priority rules; Staff

Infrastructure installations

Signaling installations; Signaling installations at level crossings; Telecommunication installations; Power supply equipment; Track; Structures; Staff

Civil engineering causes

Planned construction work; Irregularities in execution of construction work; Speed restriction due to defective track

Causes of other Infrastructure Manager (IM)

Delay caused by next IM; Delay caused by previous IM

Commercial causes

Exceeding the stop time; Request of the Railway Undertakings (RU); Loading operations; Loading irregularities; Commercial preparation of train; Staff

Rolling stock

Roster planning / re-rostering; Formation of trains by Railway undertaking; Problems affecting coaches (Passenger transport); Problems affecting wagons (Freight transport); Problems affecting power cars, locomotives and railcars; Staff

Causes of other RU

Delay caused by next RU; Delay caused by previous RU

External causes

Strike; Administrative formalities; Outside influence; Effects of weather and natural causes; Delay caused by external reasons on the next network

Secondary causes

Dangerous incidents, accidents and hazards; Track occupation caused by the lateness of the same train; Track occupation caused by the lateness of another train; Turn-round; Connection

Such classification is useful for developing an understanding of the system and identification of possible delay sources for their prevention. Many of the sources may be prevented, for example by maintenance (such as technical failures), but remain to be difficult to anticipate, although there is plenty of research dedicated to those problems. Also, serious accidents cannot be anticipated, yet often have an enormous impact on the network. Such occurrences cannot be used for delay prediction until they happen and cause a disruption in the network. Only secondary delays due to these events therefore can be later forecasted, possibly with consideration of the delay source character which implies a certain reaction in the system.

Due to the nature of railway, where the vehicles are strictly bound to the physical network, their position can be determined relatively precisely and the travel behavior is predictable, propagation of known delays is a relatively straightforward problem, although still burdened by a large extent of stochasticity. On the other hand, anticipation of a primary delay source and its impact is generally more difficult. If primary delays were simple to be predicted, perhaps they would be prevented. Prior identification of the specific delay sources is as difficult task. Instead, the listed delay sources can be used for detection of parts of the railway system that attention should be paid to and where warning signals should be searched for. Large amount of research has been carried out to understand relationships between delays and certain definable characteristics of the system, the network or exogenous elements. As an example of such factors, we can list the actual train position or realized running times (Cerreto, Nielsen, Nielsen, & Harrod, 2018; Corman & Kecman, 2018), train category, scheduled arrival time, percent of journey completed, distance traveled, time traveled, headway (Marković, Milinković, Tikhonov, & Schonfeld, 2015), time of the day (peak/off- peak), day of the week, rolling stock (Cerreto et al., 2018), traffic volume, mean speed among trains, route length, share of passenger trains exceeding certain length, traffic heterogeneity, interstation distance (Lindfeldt, 2010) and more. Although those factors are not directly delay sources, they help to identify locations, times, conditions etc., associated with a certain delay pattern. Uncovering such patterns then can be used for delay prediction.

2.3. Passenger railway delay prediction: Previous work

A large amount of research have been devoted to developing the understanding of delay behavior in the railway system. Naturally, all researchers chose a slightly different approach or a point of view. Selecting classifications relevant to our work, the approaches could be generally classified as descriptive or predictive, microscopic (on a network element or individual train level) or macroscopic (on a network level) and focused on primary or secondary delays. Reviewing the literature in a chronological order, descriptive methods dominated in the farther past while research of predictive methods begin to intensify in the last decade or two. The other classifications are represented relatively equal in the literature, but mostly with emphasis on only one category of each class (i.e. either microscopic or macroscopic combined with focus on only primary or only secondary delays).

Presenting the reviewed literature in a chronological order and beginning in a relatively distant past, it was mainly statistical and analytical methods that were applied to understand and possibly predict delay development in railway. Carey & Kwieciński (1994) put a link trip time into a context with a headway between consecutive trains on that link and developed a stochastic approximation method to estimate knock-on delays which they eventually tested and calibrated by a stochastic simulation of interactions among the trains. Applied in a network with 2-aspect and 3-aspect signaling, they found that the trip time of the following train across a link can be approximated as the maximum out of either the free running time of the following train, or the free running time of the leading train reduced by the departure headway between the two trains conformed by an adjustment constant which is optimized by a regression method.

Moving forward in time, Huisman & Boucherie (2001) developed an analytical queue-based model which returns a prediction of knock-on delays caused by differences in speed of trains using the same infrastructure segment. Their aim is to capture effects of operating different services as there is a risk of for example a faster train being caught behind a slower one. They tested and confirmed an effect of the number of trains using an infrastructure link (higher number of trains leading to a higher probability of delay propagation), and an effect of heterogeneity of the services (higher number of regional slower trains leads to an increased probability of increased travel times of the faster trains). Those scenarios considered the free running times to be deterministic. In another scenario, they introduced a random element representing stochastic nature of delays induced by primary delays such as prolonged dwell times, which resulted in significant propagation of the primary delays from the regional trains.

Yuan et al. (2002) presents a stochastic model using Monte Carlo sampling method to predict a distribution of departure delays. The prediction, which is done for already delayed trains only, is based on distributions of arrival and dwell delays: exponential distribution and normal distribution respectively. Later on, Yuan (2007) published work on how to deal with stochastic dependency in train delay and delay propagation modelling, where he defines categories of the stochastic dependency of train delays. Those are either between the arrival delay, process delay and the departure delay at a certain location; between the arrival delay and the process delay; or between previously endured knock-on delay and the arrival delay. Based on the defined dependencies, a probabilistic model was built to assess the quality of the resulting arrival and departure delay prediction.

An introduction of machine learning approach in public transport delay prediction was found in an article by Peters et al. (2006) who presented a rule-based system modelling knock-on effects consisting of the present delay information, dependency on other trains and track conflicts, and compared the results to predictions made by a neural network model built for the same task. The neural network model was built on known delay patterns where each neuron represented a delay of a certain train at a specific location. The results of the neural network model predicting delay of a particular train were concluded to outperform the rule-based system. Nevertheless, there is a sensible skepticism towards machine learning methods applied in this field and the authors recommend an extensive further research in this direction. Another application of neural network model in the delay prediction problem was by Yaghini et al. (2013). Their research focused on comparison of a variety of the model's architecture. The factors used for the prediction were an identification of an origin-destination pair, an identification of a corridor, and time describing parameters including day, month and year. The authors claim they reach high accuracy in the predictions made; however, it is unfortunately not clear how the accuracy was measured.

Besides a method of delay recognition, Hansen et al. (2010) present a model forecasting running times towards the upcoming station. They use topological description of the network, the train's present position, known dispatching decisions and a route conflict identification tool to predict running times and potential conflicts among trains. In a statistical analysis, the authors found a dependency between the present delay, and the running and dwell times. Similarly, there was a dependency found between the running and dwell times, and the period of the day.

In the same year, Goverde (2010) published an article about a timed event graph based algorithm to forecast delay propagation caused by an initial primary delay through the whole railway network. Delay propagation in the whole network was also a subject of work Berger et al. (2011). Their stochastic model of delay propagation is applicable for any public transport network and provides prediction of arrival and departure times. The main factors applied in the model are waiting policies, driving time profiles and buffer times.

A part of research by Kecman (2014) was dedicated to development of a data-driven model to predict event times in real-time in the whole network using a knowledge of the current train positions, actual delays, and the realized running and dwell times. That is according to Kecman a novel approach as he mentions that most of the scientific research preceding his work is focused on static forecasts that do not consider the present state of the network, i.e. the traffic conditions. An interesting element of the research is an online corrective feature that monitors the prediction errors and adjusts the predictions if large deviations are identified. The author, accompanied by other researchers (Kecman et al., 2015), published also an article about a Markov chain-based method of modeling probability distribution of an arrival delay. Goal of their research was to reflect the stochastic nature of delay development along with the prediction time horizon and the on-line collected and updated data.

A data driven method of delay prediction was proposed by Wang & Work (2015) who developed two regression-based models. The first model uses historical data collected on the route to predict a train delay in the upcoming stations before the train begins its trip. The second model uses data collected during the current trip and delay information of trains using the same part of the network. The models were tested on a case study and further compared to a timetable-based prediction. The first model lead to an improvement of RMSE of the delay estimation by 16% and the second model by 60%. The work therefore shows a significant improvement when real-time data are included.

So far, the majority of the presented articles proposed methods working with data related to train movements and the network state only. Work of Oneto et al. (2016, 2017, 2018) is largely innovative as they include a new exogenous factor: weather. Their data-driven approach to train delay prediction was developed over the course of three years. They begin with a comparison of multivariate regression, kernel methods, ensemble methods and feed-forward neural networks. Their research then continues by exploiting the multivariate regression model and shallow and deep extreme machine learning. Each of the models is built on historical event realization data complemented by exogenous weather data. In the latest work, the authors advanced with their shallow and deep extreme machine learning method, however, they left out the weather factor. The authors emphasize potential of all the delay prediction methods they proposed, however, following their research development, they advanced the furthest with the shallow and deep extreme machine learning method. In the future work, they plan to reintroduce weather data and propose to include also passenger flow or rolling stock conditions.

Some of the most recently published studies about train delay prediction proposed the use of Bayesian networks. Corman & Kecman (2018) presented a model for dynamic delay propagation forecast using historical and on-line registered realization data, defined dependency among trains using the same network segments, and train dependency due to guaranteed passenger transfers. Lessan et al. (2019) then propose three variations of Bayesian networks model which they use for train delay forecast. They conclude that the best performing model is a hybrid Bayesian networks model enriched by expertise of involved parties.

Table 5 Literature review: Delay prediction in railway

Reference	Method applied	Main focus	Main factors considered
Carey & Kwieciński (1994)	Stochastic approximation	Delay propagation approximation	Free running link trip times of two consecutive trains, departure headway
Huisman & Boucherie (2001)	Queuing principle based stochastic model	Delay propagation prediction; Dependence between consecutive arrival, interarrival and service times	Running time distributions
Yuan et al. (2002)	Stochastic model	Delay propagation in stations; forecast of departure delay distribution	Distributions of late arrival delays and dwell delays
Peters, Emig, Jung, & Schmidt (2006)	Neural network, Rule-based deterministic system	Delay predictions for real-time delay monitoring	Train's own delay, Dependency on other trains, Track conflict
Yuan (2007)	Probabilistic model	Delay propagation in stations prediction	Distribution of arrival times at a station, running times, clearance times on the inbound route, already suffered knock-on delays and arrival delays
Hansen et al. (2010)	Data mining	Delay recognition and classification; delay prediction towards the next station	Track occupation
Goverde (2010)	Discrete algebra, Zero-order dynamics	Delay propagation computation	Timed event graph of a scheduled railway system
Berger, Gebhardt, Müller-Hannemann, & Ostrowski (2011)	Stochastic model	Delay propagation prediction; Departure and arrival events prediction in PT	Waiting policies, driving time profiles, departure time, buffer times, train category, track conditions
Yaghini et al. (2013)	Neural Network	Comparison varying model architecture and input definition	Corridor, Day, Month, Year, O-D pair
Kecman (2014)	Graph based model	Monitoring and traffic state prediction, Rescheduling models for network-wide traffic control	Log files
Kecman et al. (2015)	Markov stochastic process	Delay uncertainty; Delay probability distribution prediction	Arrival and departure events; Current delay
Wang & Work (2015)	Regression	Using historical and real time information	Station code, Scheduled and Actual arrival and departure day and time
Oneto et al. (2016)	Kernel Regularized Least Squares; Random Forests; Neural networks	Time series forecast problem; train movements represented as events in time	Date, Train ID, Checkpoint ID, Checkpoint Name, Arrival Time, Arrival Delay, Departure Time, Departure Delay and Event Type, Forecasted weather (temperature, relative humidity, wind direction, wind speed, precipitation, pressure, solar radiation)
Oneto et al. (2017)	Deep and Shallow Extreme Learning Machines	Dynamic train delay prediction system	Traffic management data and weather condition data
Oneto et al. (2018)	Shallow and Deep Extreme Learning Machines	Data-driven Train Delay Prediction System for large-scale railway networks	Date, Train ID, Location, Event type, Day of the week, binary holiday indication, dwell times, running times, dwell times and the running times for all the other trains sharing an infrastructure section during the day
Corman & Kecman (2018)	Bayesian networks	Delay propagation prediction; Time dynamics; Probability distributions of event delays	Historical traffic realization data
Lessan et al. (2019)	Bayesian networks	Comparison of different delay prediction model schemes	Operation data (departure and arrival events)
Nair et al. (2019)	Ensemble of random forest, kernel regression and a mesoscopic simulation	Nationwide delay forecast	Network traffic states, weather condition, work zone information; Train specific dynamics; Travel and dwell time variation, track occupation conflicts, train connections and rolling stock rotations

Finally, a recent article by Nair et al. (2019) presents an ensemble method for network-wide train delay prediction. The method consists of three models. The first model is a random forest model and its purpose is to incorporate network traffic states (e.g. conflicts, present headways, weather or track maintenance). The next model, kernel regression, integrates train-specific dynamics. The last model is a simulation model which accounts for fluctuations of dwell and travel time, track occupation conflicts train connections and rolling stock rotations.

Train delay prediction and delay propagation prediction have largely evolved in the past. A shift towards data-driven approaches is apparent in the last years. The most recent studies applied neural networks, Bayesian networks, shallow and deep extreme machine learning, random forest or ensemble methods. Despite the variety of methods applied, the factors used for the predictions were rather repetitive: timetable characteristics (e.g. buffer time), train trip characteristics (e.g. train event times, travel and dwell times, present delay), train interdependencies (e.g. track occupation, headways, train connections) or network characteristics (e.g. location). Besides those, some other employed factors were temporal identification, origin-destination (O-D) information, rolling stock characteristics or track maintenance information. All directly related to the railway system and its operations. Only few methods employed any exogenous factors such as weather data. In addition, the earlier studies worked mostly with historical data, their analysis and the factors distributions, while in the recent studies, there is a shift towards usage of on-line registered data. From timetable quality assessment, delay predictions therefore advanced to on-line monitoring, dispatching support tools and communicating up-to-date information to the passengers.

2.4. Selected factors for delay prediction

Three factors that shall be included in our model were defined and argued in the previous chapter, Sections 1.1.2 and 1.2. Those are factors reflecting effects of passenger crowding, weather conditions and train interactions. Argumentation why the factors were selected focused mainly on a research gap. In the following paragraphs, we deliberate their relationship to and role in the railway system and train delay development.

2.4.1. Passenger crowding

Passengers are evidently the core element of passenger railway operations. The system can be assumed to be designed to accommodate their needs and expectations as well as possible. However, passengers, being users of the system, do not optimize their behavior with respect to the system's needs. In the time of an interactions between them and the system, they can therefore cause deviations from the system's optimal behavior. In the context of our work, the main interaction time is when passengers board and alight from a train which takes place during the train's dwell time. In scheduled train operations, the dwell time at each location is predetermined, possibly with some extra time to mitigate the dwell time deviations to some extent.

The major determinant of length of the dwell time was argued by San & Mohd Masirin (2016) to be passengers volume. Next, the dwell time also depends on whether there is a mixed flow of boarding and alighting passengers, the number of passengers standing inside the vehicle, or whether there are passengers unfamiliar with the location, impaired passengers or passengers traveling with luggage. Next to passengers characteristics, the stations characteristics also have an effect on the dwell time. Those may be a width of the platform, capacity and location of the stairs, presence of lifts or escalators etc.. Similar factors influencing dwell time were found also by Cornet et al. (2019) who developed a data-driven approach to estimate the impact of passengers volume on the dwell time and to determine the minimum dwell time defined as "*the shortest amount of time a train is required to dwell in station for the alighting and boarding process to be able to complete (including door opening and closing)*" (Cornet et al., 2019, p. 350) which is proposed to be used, for example, for network capacity assessment. They claim that boarding generally takes more time than alighting due to the necessity to climb up a few stairs.

Significance of the role passenger counts play in delay development is proven as well, for example by research of Cerreto et al. (2018), who claims that considerably high or low numbers of passengers boarding or alighting a train can increase and, respectively to some extent, decrease delays. Similarly, research by Olsson & Haugland (2004) proved there is a correlation between punctuality of departures and arrivals, and passenger counts and occupancy ratio. Effects of passenger crowding were also studied for example by Jaiswal, Bunker, & Ferreira (2009) in the context of busway operations. Increased passenger crowding was proven to increase the interaction time between the passengers and the bus at the station and therefore led to increased dwell time and eventually delay development. Next, M. Chen, Liu, Xia, & Chien (2004) proposed a method using Automatic Passenger Counter (APC) data to estimate arrival times of a bus to the next station. The method they developed consists of a Neural Network model to forecast the travel time between the stations, and a Kalman filter-based dynamic model to increase accuracy of the arrival time prediction based on the latest arrival information. The input to the first model contained identification of the time of the day, the day of the week, the weather conditions and the APC data, which included the numbers of arriving and departing numbers of passengers from and to each station. In another publication the researchers again proved the impact of a trip pattern, the day of the week, and time of day on the travel times in bus operations (M. Chen, Yaw, Chien, & Liu, 2007). Furthermore, another factor influencing train delays development in stations was found in passengers' behavior in the railway stations while transferring, volume of transferring passengers and their heterogeneity, and the time available for the transfers (Albert et al., 2017).

Generally, despite the significance of the impact passengers have on delays in passenger railway, passengers volumes are often modelled only indirectly in delay prediction models. Such indirect incorporation is done for example by variations between weekdays or weekends, or peak and off-peak hours (Oneto et al., 2017). However, Olsson & Haugland (2004) presented correlation found between departure punctuality in passenger railway and the number of passengers (the last available number of onboard passengers related to the highest number of passengers on the relevant train), showing significant decrease in punctuality with an increased number of passengers. They also observed correlation between punctuality and occupancy ratio (number of passengers divided by the number of seats on the train), although the correlation was weaker. If necessary data exists, is available and difficulties of retrieving passenger counts (such as privacy issues or uncertainty due to fare collection often located outside the vehicles) are overcome, inclusion of passenger counts into delay prediction models certainly has a high potential. On the other hand, absence of factors directly representing passenger counts in the state-of-the art train delay prediction models suggests presence of hard-to-overcome obstacles.

2.4.2. Weather conditions

Significant influence of weather conditions on delay development in railway operations was confirmed, for example by research of Oneto et al. (2016) and Ling, Peng, Sun, Li, & Wang (2018). Oneto et al. (2016) conducted research on train delay prediction in Italy including weather data and found significant improvements of delay predictions when weather information was introduced. Weather data were retrieved from the nearest weather stations and included historical data, current state and weather predictions in a desired time horizon including historical weather predictions. Correlation between train delay and its exposure to bad weather was also examined and proven by research applied in China, carried out by Ling et al. (2018). Although this research focused on conditions relatively irrelevant to the climate in the Netherlands as the weather conditions considered were blizzard, heavy snow and heavy rain, which were proven to have an influence on train delay. Strong wind, which is more relevant for the Netherlands, was however rejected as it did not show any significant correlation with train delays. That is however in contradiction with findings of Xia, Van Ommeren, Rietveld, & Verhagen (2013) who researched the role of weather in punctuality and cancellation in railways in the Netherlands. They used weather measurement data from The Royal Netherlands Meteorological Institute (KNMI), a punctuality and cancellation dataset from the major train operator NS and a national infrastructure dataset identifying infrastructure failures obtained from the infrastructure operator ProRail. The conditions they focused on were wind speed, temperature, precipitation and snow. For example, wind speed stronger than 26 m/s appeared to cause an increase in the number of disruptions by 27 %, although significant increase of the number of disruptions could be observed already from wind speeds of 19 m/s and higher. Significant correlation was also found between presence of snow cover or presence of leaves (explanation how this information was obtained is missing in the article). A trend of an increased probability of a disruption compared to the measured temperature is observable in Figure 2, showing a significant increase in the number of disruptions with extreme temperatures either below -3 °C or above 23 °C. Xia et al. also present correlation they found between the number of disruptions and observed punctuality, which they prove to be significant and negative. The retrieved effect of temperature on punctuality can be seen in Figure 3. Overall, they concluded that a significant correlation can be observed between the number of infrastructure disruptions or punctuality and wind speed, snow, precipitation, temperature and presence of leaves. To complement the conclusion of Xia et al., in the years 2002 – 2004, weather conditions caused between 4.4% and 4.7% railway infrastructure failures in the Netherlands (Daamen, Houben, Goverde, Hansen, & Weeda, 2006).

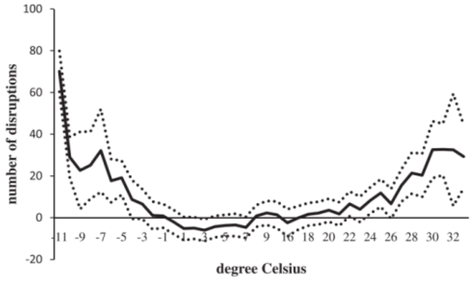


Figure 2 Effect of temperature on daily infrastructure disruptions. Dotted line represents 95% confidence interval. (Xia et al., 2013, p. 100)

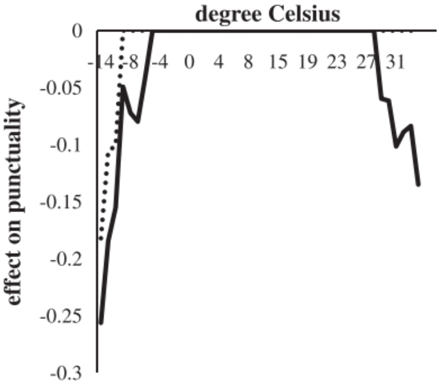


Figure 3 Effect of temperature on punctuality. (Xia et al., 2013, p. 101)

Comparable results were obtained by Palmqvist, Olsson, & Hiselius (2017) who observed the effects of various factors influencing punctuality in passenger railway in Sweden. Among those factors, four weather conditions were considered: temperature, wind speed, snow depth and precipitation. Focusing on the variables possibly relevant in the context of the Netherlands, temperature between 0°C and -5°C lead to a decrease in punctuality (compared to an average punctuality) of approximately 7.5 %-points and a similar tendency was found in high temperatures above 23°C (decrease of punctuality by 5%-points) or above 27°C (decrease of punctuality by 26%-points). They observed a decrease in punctuality by 9%-points when wind speed exceeding 23 m/s was registered, and by 2% with wind speeds above 10 m/s. The effects of precipitation on punctuality were measured in the means of an accumulated precipitation across the train's journey which showed a decrease of punctuality by 1.8%-points when 30 mm or more precipitation was accumulated, and by 2.0%-points when more than 100 mm was accumulated. Although snow depth shows a significant effect on punctuality, the large difference between the climate in Sweden and in the Netherlands diminishes its relevance for our research. Nevertheless, snow and ice formations are proven to be critical conditions increasing vulnerability of the railway system in the Netherlands in the winter season and were therefore proposed to be anticipated when the average temperature drops below 0°C and relative air humidity exceeds 80% (Neves, 2017).

In the context of delay sources, Lee et al. (2016) classify weather conditions as a train-related external factor, which is also how weather conditions were treated in the majority of the reviewed literature. However, it is important to acknowledge that weather conditions as a delay influencer are, to some extent, interconnected with passengers volumes, as weather conditions have an influence on mode choice and thus affect the number of passengers opting for railway (Cox et al., 2006; Gatersleben & Uzzell, 2007). Moreover, besides passengers volume, weather conditions influence length of the dwell time, as the boarding/alighting times are affected (Yazdani et al., 2019). Therefore, classification of weather conditions as a purely train-related factor in passenger railway delay is not entirely exhaustive and the connection to passengers should not be forgotten.

Finally, as an illustrative summary of the impact weather can have on railway operations, we share a quote from an Annual Report of the year 2018 published by NS: "*Sometimes the weather throws a spanner in the works. On 18 January a strong storm resulted in the train traffic having to stop because the safety of passengers could no longer be guaranteed. Between April and August there were faults with the trains and the track due to the hot weather. ... On the other hand, the usual autumn decrease in punctuality was limited last year.*" (NV Nederlandse Spoorwegen, 2019)

2.4.3. Train interaction: delay propagation

A major limitation of railways is that, in general, the infrastructure does not allow passing other trains on intermediate links, such as between stations or other infrastructure parts, allowing to shift to another track. If a train does not get a movement authority due to the successive track being occupied by another, possibly delayed, train, the second train has to reduce its speed or brake completely to standstill. The unplanned deceleration and acceleration, and possibly the time of standstill, generate delay of which the length further depends on the trains' characteristics and speed limits on the track segment. Although timetables are designed to avoid conflicting routes leading to train interactions and consecutive delay propagation, train running times and dwell times are of stochastic nature, therefore the realized train movements may differ from what is planned in the timetable (Goverde et al., 2008). For example Daamen et al. (2009) and Yuan (2007) further describe delays caused by trains being caught behind another train and delays caused by trains waiting for another to arrive at a station in case of ensured transfer connections.

As the main factors influencing knock-on delays, Carey & Kwieciński (1994) point at the train control, signaling system, minimum headways between two consecutive trains and speed limitations of the trains. Huisman & Boucherie (2001) then emphasize on criticality of operations combining various train types such as regional, intercity, international or highspeed trains whose different travel speeds may contribute to forming knock-on delays when, for example, a faster train is caught behind a slower one.

Several analytical models were developed to describe the dependencies among trains sharing infrastructure. As an example, Huisman & Boucherie (2001) developed a stochastic model based on a queuing principle which takes into consideration running time distributions and as an output describes the link between the consecutive arrival time, interarrival time and the service time. Recently, Harrod, Cerreto, & Nielsen (2019) proposed an analytical closed form formulation of aggregate delay where a supplement and buffer time are used as a variable to gain understanding of their role in delay and delay propagation reduction. Their findings yielded limited power of the buffer and supplement time in the task. On the other hand, distribution of buffer times was found to play a significant role in a delay propagation model based on Monte-Carlo simulation developed by Zieger, Weik, & Nießen (2018).

A predictive model for delay propagation was presented, for example, by Yuan (2007) who proposed a probabilistic model to estimate knock-on delays of trains in stations and on links between consecutive stations considering distribution of arrival times at the stations, running times, clearance times on the inbound routes, already suffered knock-on delays and arrival delays. An interested reader may also refer to the Introduction of the paper by Yuan (2007) for an overview of earlier analytical stochastic models of delay propagation, which are depreciated by the author for only implicit consideration of knock-on delays through conflicting routes or for imprecise definition of block occupancy. Furthermore, Yuan also mentions possible use of microscopic tools for delay propagation analysis, which he however discourages due to excessive time demanded for their engagement and due to lower understanding resulting from their output.

Finally, Berger, Gebhardt, Müller-Hannemann, & Ostrowski (2011) presented a stochastic model based on event graph to forecast delay propagation. The strength of the model is in its online applicability in large scale public transit networks, allowing for dynamic updates as new delays are registered. Input to the model includes the departure time, train category, track conditions, driving profile but also waiting policies and the available buffer times. Real time stochastic delay propagation prediction was also proposed by Corman & Kecman (2018), who developed a model based on Bayesian networks.

2.5. Summary

In this chapter, we defined what a delay in railways is, what its role and consequences are and how delay prediction may help improve the service provided to passengers. A delay risk is a significant factor that passengers consider when opting for a transport mode, and furthermore, it is an additional stress-causing factor in an often already stressful environment. The delay risk relates to unpredictability and possibly unreliability, depending on characteristics of the specific system. And unpredictability is what can be addressed by delay prediction. The potential result of a high-quality delay prediction is then in increased comfort of the passengers by lowering induced stress level, consequently, improve reputation of the system and to give the passengers an opportunity to adjust their schedules and travel plans to minimize their loss caused by the delay. Above that, delay prediction also has a wide range of potential application also in railway operations.

Next, we identified sources of delays in railways and explained their relation to delay prediction. The most frequently mentioned delay sources are passenger volumes, weather conditions, technical failures or human behavior. Certainly, delays can be induced by a wide range of causes which are in general difficult to foresee. Thus, instead of focusing on the specific delay causes, they are used to identify parts of the system prone to delay infliction and factors that may indicate expected delay behavior. Following the research objective, those factors were defined as passenger counts, weather conditions and train interactions.

Literature relevant to those factors in connection to delay behavior and prediction was explored at the end of this chapter, after we presented an overview of past research efforts devoted to delay prediction. The goal of the section was to explore approaches that were used and factors that were included to gain inspiration for our work. The latest research largely inclines to usage of data-driven approaches with the goal of on-line application. The most recent studies applied neural networks, Bayesian networks, shallow and deep extreme machine learning, random forest or ensemble methods. Despite the variety of methods applied, the factors used for the predictions were rather repetitive, mostly related to the timetable characteristics, train trip characteristics, train interdependencies and network characteristics. Exogenous factors such as weather conditions appeared in a few reviewed articles, however, passenger volumes did not.

Finally, the relevance of passenger counts, weather conditions and train interactions to delay development was inspected. The application of all the three factors in delay prediction was justified, which brings us to the first research sub-question.

Research sub-question(s) to be answered:

1. *Why and how should the defined factors be included in the delay prediction model?*

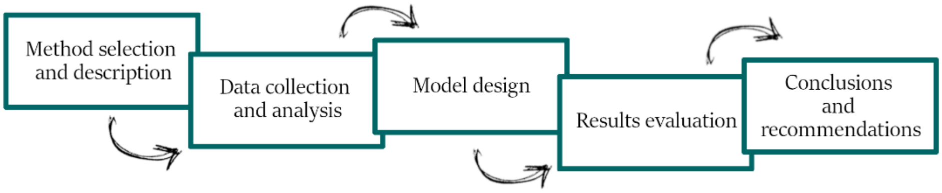
The answer to the question should be divided into three parts, each focusing on the individual factors. First, passenger counts were proven to play a significant role in delay development as the number of passengers interacting with the train during dwell time is the major factor determining the dwell time length (besides its planned length). Extended dwell time then might induce a delay which may or may not disperse along the train's route and propagate to further parts of the train's trip or even to other trains. An important finding was extracted from work of Olsson & Haugland (2004), who found correlation between departure punctuality in passenger railway and the number of onboard passengers related to the highest number of passengers on the relevant train. This served as an inspiration to us and thus scaling the number of passengers with respect to a certain representative number of passengers will be used in our work.

Next, influence of weather conditions on delay development was investigated and found to be significant. Weather conditions, that are relevant to railways operations in the Netherlands and were explored in the reviewed articles, were mainly air temperature, wind speed, precipitation and presence or risk of snow. Referring to review of delay prediction approaches, Oneto et al. (2016, 2017, 2018) included weather conditions in the form of weather forecast. Nevertheless, the weather conditions variables were in a form usual to representation of the weather conditions (e.g. temperature in °C) which is a form we opt for in our research.

Finally, delay propagation due to train interactions is a subject of large amount of research and is somehow included in majority of delay prediction approaches. Due to the nature of railways, where the vehicles are bound to the limited infrastructure, they cannot easily deviate from their trajectory to bypass a potentially conflict situation. So-called secondary delays are therefore a crucial element of delay prediction when the prediction scope implies possible existence of any interactions, as is the case of our work. Inspiration for our research then was taken especially from work of Carey & Kwieciński (1994), who point at the role of the minimum headways between two consecutive trains, and Huisman & Boucherie (2001), who emphasize on criticality of operations combining various train types.

3. Method

In this chapter, a theoretical introduction is given about the data used for our research, the method employed and finally the evaluation metrics defined and later applied. Similarly to the structure of our research approach, the generic overall scheme of the methodology can be described by the following steps:


Method selection and description
Data collection and analysis
Model design
Results evaluation
Conclusions and recommendations

This scheme is further developed and concretized in the context of our problem's nature as we apply a machine learning technique which can be regarded as a data mining problem. We drew inspiration for concretization from the general steps of the methodology from Maimon & Rokach (2005) who described a 9-steps research scheme called "Knowledge Discovery in Databases" which can be summarized as follow:

1 Developing understanding of the application domain; Goals definition.
2 Dataset creation: Data collection and integration into one dataset.
3 Preprocessing and cleansing: Dealing with missing entries, noise and outliers.
4 Data transformation: Feature selection, attribute transformation.
5 Choosing the appropriate Data Mining task.
6 Choosing the Data Mining algorithm.
7 Employing and tuning the Data Mining algorithm.
8 Evaluation and interpretation of the results with respect to the goal.
9 Using the discovered knowledge.

The first step has been already completed in Chapter 1 and Chapter 2 when the problem was introduced, and relevant literature was reviewed. This chapter provides a theoretical background and preparation for steps 2-9. Section 3.1 provides a theoretical background for the steps 2, 3 and 4. The 5th step, the data mining task selection, is theoretically explained in Section 3.2. Method selection (step 6) is presented in Section 3.3 and parameters tuning (step 7) explained in Section 3.4. Eventually, results evaluation and presentation used for answering the research question (steps 8 and 9) is explained in Section 3.5.

3.1. Data introduction

There are three main data types used as input in machine learning: *Numeric*, *Ordinal* and *Nominal*. Numeric data are represented by an integer or float number and their magnitude represents the relative distance between each other (e.g. age). Ordinal data (also called categorical) represent an order but with no meaningful distance between the values (e.g. experience level: beginner, intermediate, expert). Nominal data then lack both, an order and distance (e.g. gender) (Rokach & Maimon, 2015). XGBoost is a powerful tool for many reasons including its ability to optimize usage of the hardware available. Important to our research is that it can handle all the three data types, that it is able to overcome missing data and that it does not require data normalization.

Next, a variety of data structures is used in machine learning. The data structure we work with can be categorized as *Vectors*. A vector is a set of features with varying units and scales (Smola &

Vishwanathan, 2008). For example, in estimation of heart disease probability, the data vector could consist of age, gender, smoker identification and body mass index. All the data used in our work will be transformed into a form compatible with XGBoost, therefore into vectors associated with individual recorded train activities retrieved from the realization data. A set of characteristics captured by a vector will be referred to as a *feature set* and individual characteristics (i.e. vector elements) as *features*. The target variable, train delay, is then referred to as a label.

Although, generally, it is possible to say that the more data available the better, a too large dataset used for training the model can lead to excessively long training time. Furthermore, it is desired to avoid overfitting. Splitting the dataset into smaller instances is therefore necessary. The dataset is split into three parts, so called *training*, *testing* and *evaluation set*. The first two sets are used for tuning of the model's configuration and consecutively for the final model training. Eventually, the evaluation set is used for the model's performance assessment. If only training and testing set is used, split of the data into the two instances is proposed to be done as approximately 2/3 and 1/3 of the full dataset (Machine Learning Mastery, 2019). Due to introduction of the final evaluation set, their size was reduced, yielding the dataset being split in proportions of 0.6, 0.2 and 0.2 of the full set in the respective order. Due to the size of the dataset, it can be assumed that the samples remain representative.

Following the objective of our work a variety of feature sets is developed. All features are prepared before the models are developed. The feature sets are then created by drawing predefined sets of features from the full feature set, and they are designed in an attempt to cover a wide range of combinations of features constructed from the data. Some features are directly adapted from the data without major processing while some are designed to enhance a selected factor by applying feature engineering techniques described at relevant parts of Section 3.6 and inspired for example by Anderson, Antenucci, & Bittorf (2013), Au, (2018) or Nargesian, Samulowitz, Khurana, Khalil, & Turaga (2017).

3.2. Task selection

There are two main tasks to select from in machine learning: *Classification* (which can be further divided into binary and multi-class classification) and *Regression* (Zhou et al., 2017). As the names suggest, classification tasks predict affinity to a specific class from a portfolio of classes (e.g. train arrival time being classified as 'on time' or 'late'). Regression tasks then estimate a specific value of the target variable (e.g. a delay of a train in minutes). In the RAS competition, the competitors were asked to provide prediction in both categories: *jump* and *change* being classification tasks and the *actual delay* prediction being a regression task. As all the categories refer to the same variable: a delay, regression task naturally provides the most information and the two other tasks are implicitly answered by predicting the actual delay. Motivation for distinguishing the three categories was in complexity of the prediction, binary classification of Jump being supposedly the least complex, followed by three-class classification of Change in delay, and finally regression being the most complex task. Aiming for the most precise delay prediction information, only the regression task was selected for our work. The defined classification tasks will be used as a supplementary evaluation tool as is explained in Section 3.5. However, when comparing our results to the reference models, it is important to keep in mind that derivation of the classification tasks from the results of the more complex regression potentially leads to poorer performance in the classification tasks.

3.3. Method selection

Our work largely extends on research by Van den Bulk et al. (2018) who proposed the use of machine learning based methods, which was in accordance with findings of Hellsten et al. (2018) and Nabian et al. (2018). The portfolio of methods resulting from the researches included tree-based models and neural networks. None of the results lead to a clear conclusion which of the machine learning branches has a higher potential if applied in our research. Neither is there a clear preference arising from the literature study. However, only one model was to be selected for application in our work as our attention is on the effects of varying features rather than on performance of different models. Our preference inclined towards gradient boosting tree-based method for its easier interpretability and possible features importance analysis. Selection of this method was eventually supported by findings of Van der Hurk (2019) who continued on research of delay prediction initiated in the RAS competition (Hellsten et al., 2018) and achieved better results with tree-based model than with initially used neural networks model. Therefore, the final selection of the model that was used in our work is a decision-trees-based gradient boosting method XGBoost (T. Chen & Guestrin, 2016) as used by Van den Bulk et al. (2018).

3.3.1. Gradient boosting for regression

As described by Maimon & Rokach (2005), decision trees are predictive models belonging to a group of supervised learning methods. Supervised learning refers to methods exploring relations between input and target attributes, where the target attributes are known in the learning dataset and unknown when a prediction is supposed to be made. A model resulting from the learning process uncovers relationships hidden in the data structures which then benefit the prediction of unknown target attributes (Maimon & Rokach, 2005). In the case of this project, the target attribute, hereinafter referred to as a *label*, is a future train delay.

Following the tree structure from the origin *node* through the *splits*, the terminal node with no more further splits, so called *leaf*, denotes the predicted label (Breiman, Friedman, Olshen, & Stone, 1984). Figure 4 depicts an example structure of a (regression) tree with possible splits, where the filled circles represent leaves containing the possible prediction values.

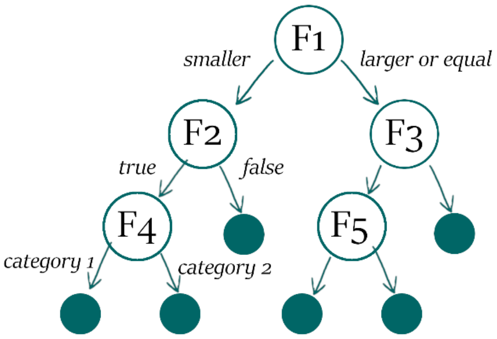


Figure 4 Decision tree structure example with splits on five features (F1 – F5).

Understandably, a single tree can have only a limited number of splits and consequentially so is the number of leaves limited. That substantially limits achievable precision of the prediction. So-called ensemble methods split the search space into multiple smaller parts. A separate model is built in each of the sub-spaces and afterwards, the models are combined into a final model following some specified rules. Boosting is one of such methods, based on sequentially building multiple relatively weak learners and combining them into a strong predictor (Maimon & Rokach, 2005). Multiple algorithms were built on this principle. One of the first ones was AdaBoost (Schapire, 2013). The algorithm builds a sequence of weak learners of various weight, each learner being dependent on its predecessor. Each tree addition is called a boosting step. The trees built by AdaBoost are typically very small, even being only so-called stems consisting of only one split and two leaves. Their weight depends on their errors. The interdependency originates in the sequential manner of the trees being build, each compensating for the errors of the preceding tree. Gradient boosting described by Friedman (2001) works on a similar principle. The main difference to AdaBoost is that larger learners are being build, and weight of all the trees is equal. The total number of trees built is either predetermined or building of the trees continues until stopping criteria are met. As the name suggests, Extreme Gradient Boosting (XGBoost) algorithm brings gradient boosting even further. XGBoost was introduced by Chen & Guestrin (2016) and keeps being further developed by many developers (XGBoost Developers, 2016a). It is one of the most powerful algorithms in machine learning domain (Chen & Guestrin, 2016) at the moment, thanks to its speed and accuracy it reaches even with problems of billions instances (XGBoost Developers, n.d.).

The essential element of the method is a regularized learning objective. A brief introduction in the mathematical formulation was retrieved from T. Chen & Guestrin (2016) where we refer an interesting reader to learn further details of gradient boosting, split finding algorithms and the XGBoost algorithm. Input of n examples can be characterized as $\mathcal{D} = \{(x_i, y_i)\}(|\mathcal{D}| = n,\ x_i \in \mathbb{R}^m, y_i \in \mathbb{R})$, where x_i is a vector of m features and y_i is the corresponding label. Output of a tree ensemble consisting of K trees can then be defined as

$$\hat{y}_i = \phi(x_i) = \sum_{k=1}^{K} f_k(x_i), \quad f_k = \mathcal{F} \qquad (3.1)$$

where $\mathcal{F} = \{f(x) = w_{q(x)}\}(q: \mathbb{R}^m \longrightarrow T, w \in \mathbb{R}^T)$ represents the regression trees' space and where q is the tree's structure and T is the number of leaves in the tree. Every f_k then refers to a corresponding tree structure q and leaf weight w, respectively w_i representing the value of an i-th leaf. Following the tree structure q, every example from the input data is classified into the leaves and the final prediction is computed by summation of the scores (given by w) from all the K trees. For learning of the functions set, the regularized objective, defined in (3.2), is minimized:

$$\mathcal{L}(\phi) = \sum_i l(\hat{y}_i y_i) + \sum_k \Omega(f_k) \qquad (3.2)$$

where $\Omega(f) = \gamma T + \frac{1}{2}\lambda \|w\|^2$. The l is a differentiable convex loss function that quantifies the difference between $\hat{y}_i$ (prediction) and y_i (true value). Ω represents a penalty given due to the model's complexity and the additional term addresses the risk of overfitting by smoothing the final learnt weights. Due to the nature of the equation (3.2) where functions are used as parameters, in every iteration, the objective is optimized in an additive fashion using a greedy algorithm to maximize the loss reduction on every additional split made. The exact definition of the loss reduction after a split and its derivation from the regularized objective can be found in Chapter 2 of introduction of XGBoost by Chen & Guestrin (2016).

3.4. Model Tuning

Configuration of a model is a crucial element in maximization of the model's performance in the context of speed and prediction accuracy. Configuration of XGBoost is done through a variety of parameters which can be divided into three groups (XGBoost Developers, 2016b):

General parameters define which '*booster*', defined as a "*model of XGBoost, that contains low level routines for training, prediction and evaluation*" (XGBoost Developers, 2016c), is used, offering either a tree-based booster (used in our research) or a linear booster.

Task parameters define the learning scheme. Two main parameters have to be set: the learning objective and the evaluation metrics. A variety of pre-defined learning objectives and evaluation metrics is offered in the XGBoost package (XGBoost Developers, 2016b). Nonetheless, selection of the parameters for our application was adopted from the XGBoost model developed by Van den Bulk et al. (2018). The learning objective used in our work thus is regression with squared loss and the evaluation metrics is a root mean squared error.

Hyperparameters dependent on the selected booster and the following text refers to those used in a tree-based booster. Most hyperparameters that can be set in XGBoost deal with the risk of overfitting and can be divided into two groups: addressing model's complexity (max_depth, min_child_weight and gamma) and increasing robustness towards noise (subsample, colsample_*) (He, 2015). The basic rule to avoid overfitting is to include only as few features as is the necessary minimum and to contemplate about complexity of the model (Galappaththi, 2015; Hawkins, 2004). Nevertheless, that is the task of the hyperparameters in XGBoost as they help to reduce the risk of overfitting and to find a balance between bias and variance, and accuracy and simplicity (He, 2015). To make the model more conservative and therefore more robust towards overfitting, one can: set a lower value to the *learning_rate* (eta) which decreases new features' weights at the end of each boosting step, controlling the boosting process; set a larger *gamma* value which causes a higher minimum loss reduction is required for an additional split on the tree's leaf node; decrease the maximum allowed tree depth (*max_depth*) which also saves computational memory; set a larger *min_child_weight* which equals to the minimum number of observations falling into each node in the building process; set *subsample* to a value other than 1 as it selects a random subset of data of a proportion size equal to the value in each boosting iteration; increase lambda or alpha value which refer to penalty terms in L2 and L1 regularization respectively (a reader interested in details about L1 and L2 regularization may refer to Ng (2004) or Nagpal (2017) for a brief overview); and finally set a lower value to *colsample_** hyperparameters especially when dealing with large numbers of features as these hyperparameters are a subset ratio of the columns used on the level of a tree (*colsample_bytree*), tree's levels (*colsample_bylevel*) and nodes (*colsample_bynode*) (XGBoost Developers, 2016b). See the default values and possible ranges of the hyperparameters in Table 6.

Table 6 XGBoost hyperparameters (XGBoost Developers, 2016c)

Parameter	Range	Default value	Selected for optimization
max_depth	[0, ∞]	3	yes
learning_rate (eta)	[0, 1]	0.1	yes
n_estimators	ℕ	100	yes
min_child_weight	[0, ∞]	1	yes
gamma	[0, ∞]	0	no
subsample	(0, 1]	1	yes
colsample_bytree	(0, 1]	1	yes
alpha	[0, ∞]	0	no
colsample_bylevel	(0, 1]	1	no
colsample_bynode	(0, 1]	1	no
lambda	[0, ∞]	1	no
scale_pos_weight	[0, ∞]	1	no

Clearly, values of the hyperparameters have a significant impact on performance of the model and as the authors of the XGBoost algorithm say, there is no universal optimal set of the hyperparameters nor an algorithm to find it (He, 2015). Available scientific literature, with information about XGBoost hyperparameters tuning, is very limited and most available sources are rather online articles shared in a machine learning community. To list a few examples, there are guidelines for manual hyperparameters tuning (Jain, 2016), randomized or full grid search (Kaggle Participant, 2017), or more advanced approaches such as Bayesian hyper-parameter optimization (Yufei Xia, Liu, Li, & Liu, 2017), coordinate descent (Restrepo, 2018) or genetic algorithm (Jain, 2018). As many of the sources lack scientific credibility, any conclusions about performance of the algorithms must had been taken with caution and rather as an inspiration than a dogma.

Only a subset of hyperparameters was selected to be optimized for application in our research, leaving the rest either in their default setting or in a defined state. See the hyperparameters selected for optimization in Table 6. The selection of the subset was based on work by Van den Bulk et al. (2018) as our work fundamentally follows upon theirs. Van den Bulk et al. fully enumerated allowed values for each of the parameters and performed a full grid search. In total, they tested 405 combinations of the varying hyperparameters. Their results showed that the optimal hyperparameters were substantially varying across the allowed value ranges among the tested models. To allow a wider range of hyperparameters values whithout increasing computational complexity, we decided to utilize a genetic algorithm (Whitley, 1994) for hyperparameters tuning. Selection of the algorithm was inspired by promising results in XGboost hyperparameters tuning application in work of Jain (2018).

3.5. Evaluation and interpretation of the results

A part of the objective of our work is to compare performance of a variety of feature sets. Each feature set is used as an input to every individual model of all the train series: having *N* train series, *N* models are built and estimated for every feature set. In sake of concise result presentation and analysis, the results are first analyzed on the level of feature sets and only at the end, we zoom into the level of individual train series focusing on extremes, contrasts and peculiar cases (see the illustration of the evaluation levels in Figure 5). Furthermore, it is inspected whether it is possible to select an 'optimal' feature set universally for all the series, or if the delay prediction on the feature set level reaches better performance with varying feature sets among the train series. Important to clarify, aggregated results across all the train series are obtained by combining the predicted values along with the relevant true values from the evaluation instance from all the train series. The overall results therefore reflect on the individual predictions, not summarized results of the predictions done for the train series.

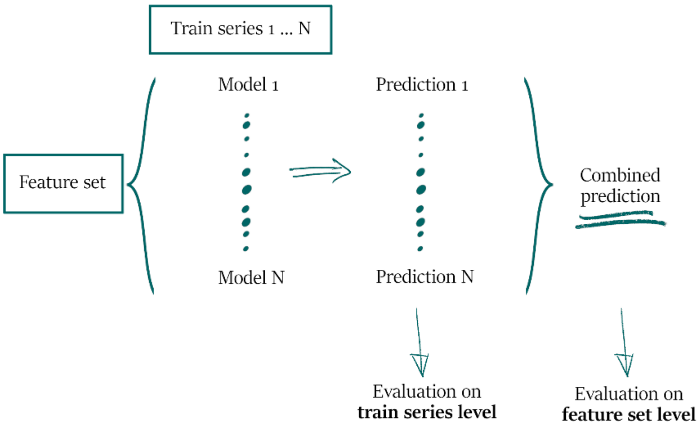


Figure 5 Evaluation levels.

3.5.1. Key performance indicators

The task of our models is regression and the evaluation metrics used in XGBoost is root mean square error (RMSE). The first determinative metrics therefore is RMSE complemented by a mean square error (MSE) and 95% confidence intervals (CI). Using solely RMSE as a performance indicator is considered insufficient for its lacking unambiguity, for example by Willmott & Matsuura, (2005) who prioritizes MSE instead. Nevertheless, their opinion is opposed by Chai & Draxler (2014) who advocate for using also RMSE in science. As there is no clear preference towards either of the two metrics, we report them both in our research. Above that, we use the metrics for comparison of a variety of models rather than for drawing conclusions about the individual models. Thus, there is no difference in using RMSE or MSE in a large part of our research. Furthermore, they are complemented by a weighted RMSE (RWMSE) as was defined in the RAS competition assessment. See the definition in the following sub-section.

Although RMSE and MSE are essential tools for assessment of the results given by a regression task, they do not provide information about the error distribution (such as reflecting the actual or predicted delay length) neither about the models performance for example with respect to criteria defined by the classification tasks in the RAS competition. Using those classification tasks as additional performance indicators, the regression results are converted to the corresponding classes by computing the predicted delay change as a difference between the predicted delay and the corresponding present delay, and consecutively assigning to the relevant classes. Finally, the two classification tasks are complemented by an additional one. That is a prediction whether there will be any delay or not, while delays up to 2 minutes count as no delay. This classification follows work of Fioole (2018b). All the derived classification tasks will be evaluated using a confusion matrix and the resulting metrics: accuracy, precision, recall and F-score; all explained bellow.

In addition, the regression results are discretized into 1-minute intervals on a scale between 0 (and less) and 15 (and more) minutes and a confusion matrix and the resulting scores are computed and presented in a suitable form. Eventually, performance of the models is further analyzed in the context of the role of the individual features within the feature sets using feature importance metrics defined in the XGBoost package (XGBoost Developers, 2016c) as explained below.

Regression: MSE and Confidence intervals, RMSE, RWMSE

MSE and RMSE are calculated as in (3.3) and (3.4) respectively, where N is the number of observations, z_f is a vector of the forecasted values and z_o is a vector of the observed values. In the context of our research, the unit of RMSE is minutes. The MSE is complemented by its corresponding 95% confidence interval (CI). To determine the CI of the MSE, the normal distribution of the squared errors is required. A normality test defined as $k_2 = s^2 + k^2$, where s is the z_score of a skew test, and k is the z_score of a kurtosis test (Bai & Ng, 2005), is used. The hypothesis that the sample is normally distributed is rejected if $p < 0.05$ (Biau, Jolles, & Porcher, 2010). In case the hypothesis is rejected, the data is resampled using a bootstrap method (Hesterberg, 2015; Kesar Singh and Minge Xie, n.d.; Pek, Wong, & Wong, 2017). An algorithm to obtain the bootstrapped MSE along with its 95% confidence intervals was adopted from Good (2006).

$$MSE = \frac{\sum_{i=1}^{N}\left(z_{f_i} - z_{o_i}\right)^2}{N} \quad (3.3)$$

$$RMSE = \sqrt{MSE} \quad (3.4)$$

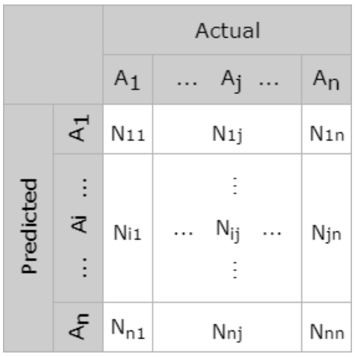
In addition, root weighted mean square error (RWMSE) was adopted from the assessment indicators used in the RAS competition retrieved from the submitted work of Nabian et al. (2018). The RWMSE is defined as in (3.5) where there is an additional variable, the weight w_i defined in (3.6) which gives more importance to errors associated with larger observed delays.

$$RWMSE = \sqrt{\frac{\sum_{i=1}^{N} w_i\left(z_{f_i} - z_{o_i}\right)^2}{N}} \quad (3.5)$$

$$w_i = \begin{cases} 0.2 & if \;\; z_{o_i} \in [-1,1] \\ 0.8 & if \;\; otherwise \end{cases} \quad (3.6)$$

Confusion Matrix

The confusion matrix (Figure 6) represents number of instances that fall into bins of combinations of predicted and actual (true) values. The diagonal then represents correctly predicted classes. The classification tasks used in our research can be described as binary and multi-class classification. In both cases, one and only one class is predicted for each instance. The sum across the matrix therefore equals to the number of instances that were subject to the prediction. (Sokolova & Lapalme, 2009)



		Actual		
		A_1	… A_j …	A_n
Predicted	A_1	N_{11}	N_{1j}	N_{1n}
	… A_i …	N_{i1}	⋮ … N_{ij} … ⋮	N_{jn}
	A_n	N_{n1}	N_{nj}	N_{nn}

Figure 6 Generic confusion matrix (Deng, Liu, Deng, & Mahadevan, 2016)

The confusion matrix is built for the following tasks:

Classification task	Classes	Notation	Description
Delay existence	*Positive* *Negative*	1 0	a delay exceeding 2 minutes is predicted a not existing delay or a delay of maximum 2 minutes is predicted
Delay jump	*Positive* *Negative*	1 0	a delay jump of 4+ minutes is predicted a not existing delay jump is predicted
Delay change	*Decreasing* *Constant* *Increasing*	- = +	a delay is predicted to decrease by 2 or more minutes a delay is predicted not to change by 2 or more minutes in either direction a delay is predicted to increase by 2 or more minutes
Discretized prediction	[0,15] *minutes of delay*		Discretized prediction of delays between 0 minutes (including delays below 0 min) and 15 minutes (including more than 15 min)

At the end, for the best performing feature set(s), additional confusion matrices are done in a graphical form of heatmaps to demonstrate distribution of errors. The delays are then discretized by rounding them with precision of 0.25 min or 1 min.

Accuracy, Precision, Recall and F1 Score

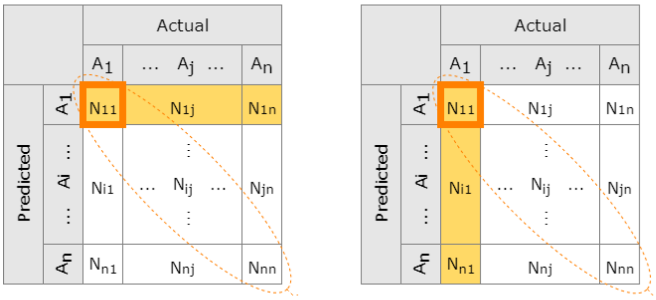


Figure 7 Precision and recall illustrations in a corresponding order (Deng et al., 2016)

Correctness of the predictions in classification tasks can be represented by accuracy, precision, recall and F1-score. Accuracy equals to the percentage of correctly predicted classes (3.7), therefore a percentage of the sum on the diagonal of a confusion matrix. Especially in a case where one class significantly predominates the other classes, accuracy may become a misleading metric. For example in the case of a delay prediction, there is no delay jump in a vast majority of the observations. Predicting no delay jump for all the instances would easily return very high accuracy. Accuracy therefore is rather disregarded in our research.

$$Accuracy = \frac{\sum_{i=1}^{n} N_{ii}}{\sum_{i=1}^{n}\sum_{j=1}^{n} N_{ij}} \qquad (3.7)$$

Instead, attention is paid to the remaining metrics. Precision represents an accuracy given a certain predicted class (3.8). Opposite to that, recall is an accuracy supposing a specific actually observed class (3.9).

$$Precision_i = \frac{N_{ii}}{\sum_{k=1}^{n} N_{ki}} \quad (3.8)$$

$$Recall_i = \frac{N_{ii}}{\sum_{k=1}^{n} N_{ik}} \quad (3.9)$$

A harmonic mean of the two metrics, precision and recall, is called F-score, or commonly F1-score, where 1 denotes that no weighting is used and precision and recall both have the same weight in the equation (3.10).

$$F1\_score_i = \frac{2 \cdot Precision_i \cdot Recall_i}{Precision_i + Recall_i} \quad (3.10)$$

The metrics will be calculated and presented for the same classification tasks as above.

Feature importance

XGBoost package offers evaluation of feature importance by global measures *'weight'*, *'cover'* and *'gain'*. Given a feature set F and a set of trees T, each tree consists of a set of splits S_t and is used by a number of observations O_t, where each observation follows a sequence of splits K_O^t. Weight is "*the number of times a feature is used to split data across all trees*"(XGBoost Developers, 2016c) and can be mathematically written as:

$$weight_f = \sum_{t=1}^{T} \sum_{s=1}^{S_t} split_s^t \quad (f \in F, split_s^t \in [0,1]) \quad (3.11)$$

,where $split_s^t$ is equal to one if the split belongs to the feature *f*, and zero otherwise. If $observations_s^t$ represent the number of observations entering a split *s* on a tree *t*, cover of a feature f can be defined as:

$$cover_f = \sum_{t=1}^{T} \sum_{s=1}^{S_t} split_s^t \cdot observations_s^t \quad (f \in F, split_s^t \in [0,1], observations_s^t \in \mathbb{N}) \quad (3.12)$$

or in words as an "*average coverage across all splits the feature is used in.*"(XGBoost Developers, 2016c). Last, gain, which is an "*average gain across all splits the feature is used in*" (XGBoost Developers, 2016c) can be mathematically defined as

$$gain_f = \frac{\sum_{t=1}^{T} \sum_{o=1}^{O_t} \sum_{k=1}^{K_O^t} split_{t,o}^k \cdot \left(loss_{t,o}^{k-1} - loss_{t,o}^k\right)}{\sum_{t=1}^{T} \sum_{o=1}^{O_t} \sum_{k=1}^{K_O^t} split_{t,o}^k} \quad (3.13)$$

$$(f \in F, split_{t,o}^k \in [0,1], loss_k \in \mathbb{R}_+)$$

where $loss_{t,o}^k$ is a value given by a loss function after introducing a $split_{t,o}^k$. $loss_{t,o}^0$ then refers to the loss function value before any split is introduced.

Important to note, only gain will be elaborated on as it is the most informative feature importance indicator for our work as is explained together with the results presentation in Section 6.5.

3.6. Summary

In this chapter, we provided a necessary theoretical minimum to understand the upcoming steps of data collection, processing and features retrieval, the model development and tuning, and finally results presentation. The data will be transformed to features which combined form feature sets. As the subject of our work is delay prediction, the task we will work with falls into the category of regression tasks. The method selected to perform the task to obtain delay predictions is a decision-trees-based gradient boosting method called XGBoost wherein several parameters will be tuned to optimize its performance. Finally, key performance indicators for assessment of the results were defined which brings us to research sub-question 2.

Research sub-question(s) to be answered:

2. *What are appropriate key performance indicators considering the model's characteristics?*

The algorithm in the model-building process evaluates the intermediate steps by the RMSE. Furthermore, either RMSE or MSE is conventional metrics for regression results assessment. Assessment solely based on either of these metrics, however, overlooks the models performance on a more microscopic level. Additional performance indicators were therefore derived from classification tasks defined in the RAS competition (delay *jump* and delay *change*) as it was assumed that those metrics were defined to reflect needs and issues arising from the practice of NS and ProRail. Next, an identification of an existing delay (a delay exceeding 2 minutes) was added following research by Fioole (2018b). Further, to observe the models' performance on the level of the delays' length, the regression results will be discretized and assessed similarly as a classification tasks using a confusion matrix and the resulting metrics: precision, recall and F1-score. Finally, the individual features and their importance in the models will be assessed using feature importance indicators provided in the XGBoost library.

4. Data

This chapter provides a detailed description of the datasets that were used for feature creation and label retrieval. First, the original form of the dataset is presented along with the data source, then, the derived features are presented and discussed. At the end of the chapter, analysis of the data is provided focusing mostly on the relationship between the various features and the target variable; the future delay.

For the sake of comparability with the research preceding our work, data used in our research substantially depended on the data provided in the RAS competition as that was the data used for development and assessment of models proposed by Hellsten et al. (2018), Nabian et al. (2018) and Van den Bulk et al. (2018). The original dataset (INFORMS, 2018c) was provided mutually by NS and ProRail and consisted of seven datasets in total, all referring to the same time period from the 4th of September 2017 to the 9th of December 2017. Two datasets from the original package were used for our work: Planned timetable and Realization data. Those datasets were complemented by data revealing train interactions through the minimum required headways provided by ProRail, passenger counts estimations provided by NS, and finally by publicly available weather conditions measurement data of KNMI, The Royal Netherlands Meteorological Institute (KNMI, n.d.-a). The datasets are presented in the respective order. As the data came from multiple sources in varying formats, considerable data processing was inevitable. For the purpose of replicability of our research, the process is described and explained in the Appendix.

4.1. Planned timetable

The planned timetable, presented in Table 7, is an overview of a generic week within the relevant time period complemented by additional information about the train characteristics and consequent attributes relevant for the train's activity. The timetable is crucial for delay prediction as it provides information about the upcoming events and train movements. Furthermore, it is possible to derive characteristics of the available attributes such as a frequency certain location is visited, how many locations a train passes through within a time window etc., which was used in our work as described later.

Secondly, as the timetable is a significantly smaller instance than the realized data, covering five days instead of a period of roughly three months, it provides a convenient structure to define patterns of train interactions using additional data in the later steps. Above that, the realized data is virtually unknown for the time to come in the prediction process and any assumption of what is to come thus must be based on the timetable.

Attributes, that were the most important for our research, were the day of the event from which the day of the week was derived, naturally the planned time, the train number along with the train series number which was derived from the pattern code, the location, the activity and the train characteristic. Further, the order number was used in data processing for correct activity order control. No international trains were included in our research and the identification of locations abroad was therefore irrelevant. Bare driving time, traction type and timetable speed were not applied in the later model as this information was assumed to be implicitly hidden in the remaining characteristics. Although, traction type may carry information about a driving pattern such as acceleration and braking curves, it was assumed that the large differences between driving patterns of train categories, e.g. IC or SPR, are of higher importance than the nuances among individual traction types. Moreover, models were developed for each train series individually and therefore traction type was generally implicit in the train number.

Table 7 Planned timetable dataset example. (see glossary in the Appendix section A.1, Table 40)

Day	Train number	Direction	Location	Abroad	Activity	Order number	Planned Time	Train Characteristic	Pattern	Bare Driving Time [s]	Traction Type	Timetable Speed [km/h]
04.09.2017	511	O	Zl	N	V	1	04.09.2017 5:44	IC	B500	-	ICM12	140
04.09.2017	511	O	Zlgea	N	D	2	04.09.2017 5:45	IC	B500	124	ICM12	140

4.1.1. Features derived from the timetable data

Three features in two version were derived from the dataset representing the upcoming activities within the time window of 20 minutes. These features were created as a simple timetable-based representation of business of the upcoming network segment. Examples of the features can be seen in Table 8, where each row represents the features belonging to an example data instance. The first version named *'Activities'* reflects the number of activities [D, V, K_V] that will come in the upcoming 20 minutes. As V and K_V refer to the same location as the corresponding V and K_V, they are not considered. The features are supposed to reflect how many locations that may be sources of delays are to come in the prediction period. This information may identify a higher probability of train interactions by the number of the upcoming locations, or a higher probability of a delay induced by an extended dwell time suggested by a high number of stations. The second version, called *'Frequency'*, goes a step further by considering whether the upcoming locations are busy in general. The feature is highly simplified, nonetheless, the value consists of a cumulative number of the times each of the D, V and K_V locations occurs in the timetable. It is assumed that when a location occurs in the timetable with high frequency, it is busy and therefore there is a greater chance of any kind of interaction with other trains or that there might be a high passenger flow which might lead to a delay gain.

Table 8 Example of values of 'Stations to come' features.

Activities			Frequency		
V	K_V	V	V	K_V	V
1	1	11	6043	2502	21277
1	1	10	6043	2502	19083
1	1	12	6043	2502	19448
0	1	7		2502	9995
0	1	8		2502	11269
0	1	6		2502	8183

4.2. 'Realization' data

The Realization data (or in other words Automatic vehicle location or AVL data), presented in Table 9, contain registrations of realized train movements at the timetable points. The data come from the period from the 4th of September 2017 to the 9th of December 2017 and consist of the traffic date, planned and realized time, train series identification and train number, train type, location and activity type. Furthermore, delay, delay jump, and a cause of the delay jump are identified. Unlike delay jump defined in the RAS competition, the Delay jump in this dataset refers to a change in delay in comparison to the previous registration of the train. This dataset is crucial for delay identification as the delay is equal to the difference between the realized and the planned time. The realized movements are essentially based on the timetable, however operational changes in the train movements introduce inconsistency with the original schedule. Trains may have been rerouted, turned around, cancelled, added etc. Train numbers then were not always correctly assigned or changed, and trains may had been driving on unexpected routes, at unanticipated times or under identification numbers that are not present in the timetable.

Table 9 Realization data dataset example (see glossary in the Appendix section A.2, Table 40)

Traffic Date	Train series	Train Characteristic	Train number	Location	Activity	Planned Time	Realization	Delay [min]	Delay Jump [min]	Cause
04.09.2017	500E	IC	512	Ut	V	04.09.2017 06:18	04.09.2017 06:18:49	0	0	-
04.09.2017	500E	IC	512	Utwa	D	04.09.2017 06:19	04.09.2017 06:20:48	1	1	-

4.2.1. Features derived from the realization data

As the delay prediction models are built for each train series separately, the train series number and train type are irrelevant information in the form of a feature and is not included in any feature set. Further, the train number is left out, as it duplicates the time features which were split into features representing an *hour* and *minutes* in the hour. The realization time was selected over the planned time to reflect the actual present state of the system. Above that, the combination of the delay and either realization or planned time implicitly include information about the other one. Furthermore, a *day of the week* was derived from the date to reflect weekly patterns in the operations while the time of the day is expected to reflect the patterns throughout the days. Next, a *direction* of the train is included as a feature, as it is expected that the patterns might differ per direction (due to different demand or driving patterns for example). The direction is represented by a binary value. To recognize spatial position of the train and its potential effect on delay development, a *location* identification number is included. The identification number was retrieved by replacing location names by unique numbers. Unlike the other features in this set, location is determined by a categorical variable which imposes a risk of misinterpretation of the feature by the model. Eventually, to reflect the recent delay development, the present *delay* of the train and the *delay at the two preceding registration points* was retrieved. An example of the base feature set derived from the realization data is presented in Table 10.

Table 10 Example of the base feature set.

Day of week	Hour	Minute	Direction	Location	Delay	Delay 1 before	Delay 2 before
1	7	45	0	125	0.25	Nan	Nan
1	7	53	0	265	1.50	0.25	Nan
1	8	02	0	342	2.00	1.50	0.25

4.2.2. Label derived from the realization data

The label, i.e. the target variable, is a delay of a train 20 minutes in the future from the last registration point (respectively the delay the train has at the last upcoming registration point within the 20 minutes time window). It was retrieved from the delay identified in the Realization data and its units therefore are minutes. The precision selected is 0.25 minute as a compromise between unnecessarily high and complex precision of delay prediction in seconds and a full-minute delay prediction which was assumed to be unsatisfactorily sparse. It is important to clarify that the 20 minutes threshold was strict, and prediction is therefore made up to the last scheduled registration point before the last registered point plus 20 minutes. Therefore, a prediction is made from the latest registration point 20 minutes to the future from that point. For example, it is 10:45:30 and delay is supposed to be predicted for a train T. The prediction will be done based on the data collected up to the last registration point which was at 10:43:20. The label used for training the model and for evaluation will refer to a delay, that the train had at the last registration point before or at 11:03:20.

It is therefore highly unlikely, that the prediction truly refers to a state that will occur 20 minutes in the future as the highest probability is that the last considered reference point will be a few minutes before that threshold. In the fictious case, it could be at 11:01:00 for example.

4.3. Train interactions

The provided dataset consisted of pairs of train series of which the trains share some infrastructure points (signals) within a relatively short period of time. Due to safety requirements, a minimum headway time is required between such pairs of trains. The headways were generated by a tool developed by ProRail called ROBERTO which computes the headway times based on infrastructure and timetable data, train characteristics, block section occupation times and signaling aspects. Pairs of trains that share a piece of the infrastructure are selected and virtually moved closer to each other until a hindrance occurs. The minimum required headway for the specific situation is then derived in the form of a minimum time gap in seconds. (Middelkoop, 2010)

The dataset (see example in Table 11) consists of pairs of train series, where the 'Train 1 series' (T1) refers to the leading train while the 'Train 2 series' (T2) identify the series of the train that follows. Whether a train is leading or following should be understood in a temporal dimension as the trains may pass the infrastructure points in different directions. Rows in the dataset were by default sorted in order of the trains' movement. A sequence of rows belonging to an identical pair of T1 and T2 within an uninterrupted block of rows represented a sequence of infrastructure pointes traversed by the trains in the respective order. Such a sequence suggests that the trains follow each other also in a spatial context.

The timetable and the realization data are spatially specified on the level of the 'Areas' in this dataset. The list of signals therefore provides additional information of what specific parts of the infrastructure are shared (Signals), but it is irrelevant for application in the delay prediction model because the information cannot be transferred on such detailed level onto the realization data. Through an extensive processing of the headways data (described in the Appendix A.3 Data processing), relevant pairs of train series interacting at the timetable locations were derived. For each train series interaction, the first moment of the first train causing an interaction was registered. That, combined with the timetable data, was converted to train interactions patterns. In essence, for each timetable point of all the train series, the relevant interactions were identified. Eventually, the interactions patterns were combined with the relevant instances of the realization data.

Table 11 Minimum required headways dataset example. (see glossary in the Appendix section A.3, Table 42)

Train 1 series	Train 2 series	Train 1 type	Train 2 type	Area name	Signal name	Local min headway
2100	2100	VIRM08	VIRM08	Hlmvam	26	174
2100	2100	VIRM08	VIRM08	Hlm	80	89.65
2100	4800	VIRM08	SGMm06 (3+3)	Asd	67A	104.85
2100	4800	VIRM08	SGMm06 (3+3)	Asd	4	125.12

4.3.1. Features derived from the train interactions data

In total, 6 versions of the train interactions feature set were created. The first three versions contain features representing each of the interacting train series. The feature sets therefore consists of 82 columns which, in contrast with less than 10 columns of the basic features, is overwhelming for the algorithm. Above that, interactions are identified in a minority of the entries as the models are built and trained for individual train series separately and each train series interacts with only a small subset of other train series. Therefore, in the final feature set associated with the data relevant for a specific train series, only columns with the number of unique values higher than 1 are kept (in other words, only columns where there is at least one interaction identified) and the rest is discarded which significantly reduces the number of features and therefore increases the chance of the features having an effect on the prediction quality.

The first version is the actual *expected headway*, which was calculated by subtracting the leading train's delay from the planned headway. The assumption is that the risk of delay propagation increases as the expected headway decreases. Non-existing interactions were therefore filled by a large number, acting as a large headway.

Table 12 Example of an expected headway feature set ('Interactions' feature set version 0).

2100	2200	11600	2300	2400	11700	2600
99	99	99	9.5	99	99	6.5
99	99	99	8.0	99	99	7.5

The next version of the train interactions feature set is an *expected violated headway*, which was calculated by subtracting the leading train's delay from the planned margin (the planned headway minus the minimum required headway).

Table 13 Example of an expected violated headway feature set ('Interactions' feature set version 1).

2100	2200	11600	2300	2400	11700	2600
99	99	99	99	99	99	-1.0
99	99	99	-0.5	99	99	99

The third version contains solely the *binary identification* of existing interactions with the possible train series (see an example in Table 14). These features were derived from the 'expected headway' features by replacing the value 99 by 0 (False) and otherwise 1 (True).

Table 14 Example of binary identification of train interactions ('Interactions' feature set version 2).

2100	2200	11600	2300	2400	11700	2600
False	False	False	True	False	False	True
False	False	False	True	False	False	True

Further, each feature set was compressed into a more concise version where the columns were aggregated by the train series type: IC, SPR and LM. The expected headway and the expected violated headway were both converted by taking the shortest expected headway/expected violated headway among the train series belonging to the relevant train type category. Finally, the binary identification of train interactions was converted by summing up the number of identified interacting train series within the train type category.

Table 15 Examples of train interactions feature sets aggregated by train types ('Interactions' feature set versions 3-5).

Expected headway			Expected violated headway			'Binary' interaction identification		
IC	SPR	IC	IC	IC	SPR	IC	SPR	LM
5.0	7.5	99	99	99	99	2	3	0
4.0	6.0	8	-0.5	99	99	1	2	1

4.4. Passengers

The dataset containing information about the number of passengers traveling between pairs of stations during the relevant time period was provided by NS. The data are based on smartcard data, i.e. the numbers of passengers checked-in and -out with their chip-card or a ticket at locations in the Dutch railway network. The locations of checking-in was paired with the location of the check-out. According to the time of the travel, possible train connections between the origin and destination were enumerated and assigned with a probability that the passenger traveled by that specific train connection. Factors such as transfers, travel time or train type were considered to compute the probabilities. The final output is an estimate of the numbers of passengers travelling on each train between all the pairs of locations, and an estimate of the numbers of passengers boarding at each location. Due to high level of confidentiality that hinders sharing passengers related data, this data are confidential and cannot be shared in any way. The example numbers presented in Table 16 are fictional, solely representing the form of the dataset.

The dataset consists of the date, the train number, the pair of stations between which the passengers travelled and the corresponding times of the departure and the arrival. The departure and arrival times do not fully correspond with the arrival and departure times in the realization data. That is because the times in the realization data are retrieved from the registration points on the infrastructure when approaching or leaving the station, while the times in this dataset refer to the time of arrival to and departure from the platform. Finally, the total number of passengers that travelled by the train on the link between the two stations and the number of passengers that boarded at the origin station are specified along with the number of seats on the train that were planned in the schedule and that were actually realized. Eventually, the number of passengers that alighted from the train was calculated for each station.

Table 16 Passenger counts dataset example. (see glossary in the Appendix section 0, Table 43)

Date	Train number	Departure station	Departure time	Arrival station	Arrival time	Total number of passengers	Number of boarding passengers	Planned seats	Realized seats
1-sep-17	104	AH	01SEP2017 21:00:04	UT	01SEP2017 21:29:10	123.45678900	98.76543210	100	100
1-sep-17	104	UT	01SEP2017 21:31:50	ASD	01SEP2017 21:53:03	123.45678900	98.76543210	100	100
1-sep-17	105	ASD	01SEP2017 08:03:02	UT	01SEP2017 08:28:01	123.45678900	98.76543210	200	250

4.4.1. Features derived from the passenger data

In total, five versions of the feature set were created containing between 2 and 5 features. The first version (version 0) consists of 2 features only: the *seats ratio* and *peak hour departure* identification. The seats ratio refers to the difference in seating capacity that was planned and that was actually realized. The value is a floating number where 0 means a cancelled train, 1 means no change in rolling stock, numbers between 0 and 1 mean that a smaller train was used and numbers above 1 identify a when a train with larger capacity than was planned was used. The peak hour departure was introduced to represent the effects of increased passengers volumes in the time period. It is a binary feature where 0 means off-peak time and 1 refers to a peak hour.

With respect to the passenger counts, it is assumed that the timetable along with the planned rolling stock are planned to ensure undisturbed operations under regular conditions. If significant deviations from the normal state occur, the systems behavior may deviate as well. In the context of passenger numbers, the system is assumed to be designed for a ‘regular’ number of passengers and an unusually high number of passengers may cause boarding and alighting times longer than was accounted for and cause a delay. On the other hand, extremely low number of passengers may lead to shorter boarding times giving an opportunity to recover from a delay. It therefore is the deviation from the ‘normal’ state that brings a higher risk of a delay occurrence. The numbers of passengers that were used for designing the timetable and assigning rolling stock were unknown and were therefore substituted by a median value that is taken as a representation of the ‘normal’ state. The deviation from the ‘normal’ state was then computed by normalization of observed passenger counts to the median. The median was taken from multiple categories, each used as an individual feature set version (versions 1-4). The categories are:

Train number and location

Refers to a level of passenger movements for a train number at a specific location. The train number implicitly includes the time of the day and the direction. Day of the week is not considered. Relation to other trains of the same train series is missed.

Train number, location and day of week

Same as before but in addition, weekly patterns are considered.

Train series, location and peak hour

Considers all trains in a series regardless the exact time, but differentiating between peak and off-peak hours. Day of the week is not considered.

Train series, direction, location and peak hour

Same as above but with distinction between the directions.

The first two columns in Table 17 present the basic feature set derived from the passenger data, which includes only the seats ratio and peak departure identification. The remaining columns of Table 17 then represent an example of the feature sets reflecting the normalized numbers of passengers in the four different categories. The passenger data were evidently calculated for departures from stations and stops only. To provide the information on the intermittent locations as well, the values were copied onto all the registration points following the departure up to the arrival to the next station or a stop.

Table 17 An example of a Passengers feature set version 0 (first two columns) and versions 1-4 (all columns)

Seats ratio	Peak departure	Seats ratio	Peak departure	Total	Board	Alight
1.155	False	1.155	False	1.103	1.103	
1.155	True	1.155	True	1.121	1.146	1.068
1.155	True	1.155	True	1.121	1.146	1.068
1.155	True	1.155	True	I.14	1.586	1.142

4.5. Weather

Weather condition data were retrieved from the website of The Royal Netherlands Meteorological Institute (KNMI) via publicly available online interactive form (KNMI, n.d.-c). The data collected consisted of the date, time and location identification, and hourly measurements covering the average wind speed, the highest wind speed, the average temperature, the minimum temperature, the measured precipitation, the scaled horizontal view distance and the binary identification of presence of the mist, rain, snow, the storm or ice formations (see Table 18). The data originates from 50 weather stations in the Netherlands. However, not all of the weather stations provided all of the selected data, and some were thus disregarded. Nevertheless, each of the weather stations was identified by its number, name and a geographical location denoted by longitude and latitude, which was used to compute the great-circle distance to the infrastructure locations and to find the nearest weather station for each of the infrastructure locations. More information about the weather station selection and the later coupling to the infrastructure locations is in Appendix section A.5.

Table 18 Weather dataset example. (see glossary in the Appendix section A.5, Table 44)

Station number	YYYYMMDD	Hour	Average wind speed [0.1 m/s]	Highest wind speed	Temperature [0.1 °C]	Min temperature [0.1 °C]	Precipitation [0.1mm]	Horizontal view distance	Mist	Rain	Snow	Storm	Ice
209	20170904	1	50	60									
215	20170911	16	70	140	153		-1	70	0	1	0	0	0
215	20170911	18	70	130	135	131	31	39	0	1	0	1	0

4.5.1. Features derived from the weather data

The feature set versions were derived from the weather data. All of the variables allow a direct application in the model and did not require any transformation. As some of the variables were not measured by all the weather stations, one feature set version consists of the weather conditions measured by all the relevant weather stations (to allow coupling of the stations and the infrastructure locations with lower distances). The next feature set version consists of all the selected weather conditions and the last version combines the binary variables as a sum representing the degree of bad weather. That was done to reduce the number of features especially due to sparsity of presence of those weather conditions. See Table 19 - Table 21 for examples of the feature set versions

Table 19 Example of a Weather feature set version 0.

Average wind speed	Highest wind speed	Temperature	Precipitation	Horizontal view distance
70	120	147	-1	70
70	130	135	31	39

Table 20 Example of a Weather feature set version 1.

Average wind speed	Highest wind speed	Temperature	Precipitation	Horizontal view distance	Mist	Rain	Snow	Storm	Ice
70	120	147	-1	70	0	1	0	0	0
70	130	135	31	39	0	1	0	1	0

Table 21 Example of a Weather feature set version 2.

Average wind speed	Highest wind speed	Temperature	Precipitation	Horizontal view distance	Bad weather
70	120	147	-1	70	1
70	130	135	31	39	2

4.6. Data overview and analysis

To gain understanding of the data, basic characteristics of the data are presented. First, the main train trip attributes are briefly described, followed by observations related to the measured delays. Next, relations between the delay or delay change and a set of selected features are observed and presented. Finally, expectations of the features' impact on the prediction are derived.

79 *train series* were selected as relevant for our research. Those are train series operating regularly in the days of Monday, Tuesday, Thursday and Friday in the relevant time horizon. Furthermore, only train series operating exclusively in the Netherlands were selected. The list of the train series considered in our work is in the appendix in Table 51. The input data for the models consist of between 1,165 and 445,519 instances per train series. On average, each train series is represented by 109,575 instances. In total, 461 *locations* appear in the data with 1 to 119,381 registrations in the realization data in the measurement period. An average number of registrations is 18,859 per location. The most commonly registered *activity* is passing through ('D') with more than 4,000,000 occurrences, followed by naturally equal number of short arrivals and departures, which, however, appears in the data approximately a third of the times, and finally regular arrivals and departures that were registered a little less than 1,000,000 times. The numbers of registrations per *day* are relatively balanced, around 1,800,000, with slightly less registrations on Tuesday (around 1,300,000). With regard to the *train type*, there are almost 4,200,000 registrations of IC trains, about

4,500,000 registrations of SPR trains (including about 25,000 registrations of ST trains) and nearly 90,000 registrations of empty rolling stock movements (LM).

In total, approximately 19,250,00 minutes were driven by all the trains registered in the realization data. Out of those, approximately 1,750,000 minutes were driven above the planned time which yields 9% of the total time driven being a delay. Table 22 shows the top 15 train series with the largest proportion of delays with respect to the total driven time in both directions.

Table 22 The first 15 train series with the highest percentage of total delay minutes out of the total minutes driven within the measured period in both dorections (O=Odd, E=Even).

Series	Minutes Driven (O)	Minutes Delay (O)	Minutes Driven (E)	Minutes Delay (E)	Delay (O)	Delay (E)	Minutes Driven total	Minutes Delay total	Total Delay
20300	5 483	1 305	5 905	1 790	24%	30%	11 388	3 095	27%
5300	29 577	6 796	0	0	23%		29 577	6 796	23%
2600	47 999	11 470	53 207	11 180	24%	21%	101 206	22 650	22%
2800	67 225	9 336	69 224	11 730	14%	17%	136 450	21 066	15%
6100	28 929	3 870	28 003	4 560	13%	16%	56 932	8 430	15%
1500	96 741	13 480	92 622	12 220	14%	13%	189 363	25 700	14%
15800	137 058	20 530	146 410	17 470	15%	12%	283 467	38 000	13%
4700	52 282	6 429	50 092	6 286	12%	13%	102 374	12 715	12%
28300	8 567	1 071	8 123	949	13%	12%	16 690	2 020	12%
3400	9 927	1 052	10 409	1 345	11%	13%	20 336	2 397	12%
6300	101 393	12 390	100 253	11 070	12%	11%	201 646	23 460	12%
8900	15 557	1 335	14 515	2 142	9%	15%	30 072	3 477	12%
6700	4 732	635	4 926	472	13%	10%	9 658	1 107	11%
7500	32 556	4 596	33 406	2 950	14%	9%	65 962	7 546	11%
7300	31 702	3 090	33 516	4 332	10%	13%	65 218	7 422	11%

The majority of delays is between -1 and 1 minute, specifically 58.3% of all the delays registered in our data fall in this range. Generally, a negative delay, thus driving ahead, occurred in 17.6% of the train registrations. It is important to keep in mind that multiple registration refer to the same train trip and, for example, many registration points shortly after each other can affect the statistics. Next highly represented delay length interval is between 1 and 2 minutes of a delay which counts 20.3% of the registrations. The counts then significantly drop as the delay increases, to 7.9% on a delay of 2-3 minutes, 3.9% on a delay of 3-4 minutes and 2.2% on a delay of 4-5 minutes. Only 5.1% of the observed delays exceeded 5 minutes, and only 1.6% exceeded 10 minutes.

In the context of a delay prediction it is logical to observe how the delay actually changes over the time span of the prediction: 20 minutes. The available data revealed that the delay decreases in 37.5% of the cases. The assumption of the Naïve forecast is that a delay is always reduced by 1 minute in the considered time span. In reality, this was the case in 22.8% observations (this does not account for the ‘duplicity’ caused by the existence of multiple registrations of the same trip within relatively short period of time that refer to the same event). In 37.1% of the observations, the delay increased by up to 1 minute, in 18.4% by 1-2 minutes and in 4.8% by 2-3 minutes. An increase of a delay by more than 3 minutes was registered in 2.2% cases.

Table 23 length and delay change extent distribution.

Distribution of a delay length				Distribution of a delay change			
interval [min]	Observed	interval [min]	Observed	interval [min]	Observed	interval [min]	Observed
[-, -3)	0,1%	[4, 5)	2,2%	[-, -5)	1,1%	[2, 3)	4,8%
[-3, -2)	0,2%	[5, 6)	1,3%	[-5, -4)	0,7%	[3, 4)	1,3%
[-2, -1)	2,0%	[6, 7)	0,9%	[-4, -3)	1,4%	[4, 5)	0,4%
[-1, 0)	15,3%	[7, 8)	0,6%	[-3, -2)	3,1%	[5, 6)	0,2%
[0, 1)	43,0%	[8, 9)	0,4%	[-2, -1)	8,4%	[6, 7)	0,1%
[1, 2)	20,3%	[9, 10)	0,3%	[-1, 0)	22,8%	[7, 8)	0,2%
[2, 3)	7,9%	[10, 15)	1,0%	[0, 1)	37,1%		
[3, 4)	3,9%	[15, +)	0,6%	[1, 2)	18,4%		

Figure 8 and Figure 9 illustrate pairs of the current delay and the corresponding change with respect to the future delay. It is obvious, that the highest variation in the change in the future delays occurs from low or non-existing delays as the future delays corresponding to the current delays of 0 minutes vary from -10 minutes up to the maximum of +45 minutes. Note that the limits of the delays are influenced by the preceding data processing, which also causes the strict upper limit for delay increase. Understandably, the maximum observed delay decrease corresponds with the current delay length. However, that applies only up to a delay of approximately 20 minutes, from where on the observed delay decreases become smaller or event negligible around the highest measured delays.

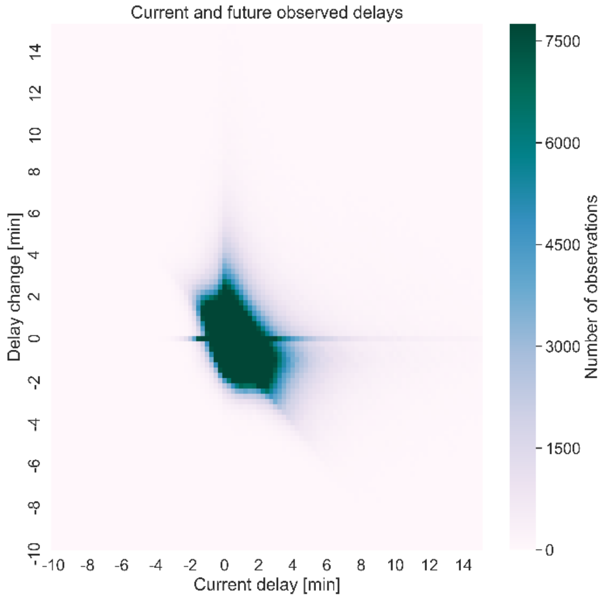


Figure 8 Heatmap plot of the current delay and the corresponding delay change.

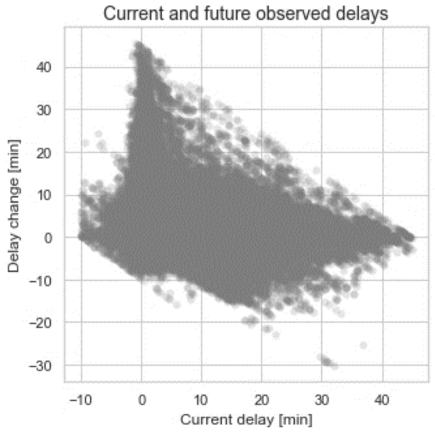


Figure 9 Scatter plot of the current delay and the corresponding delay change.

To observe if there are any recognizable patterns in the data that would suggest a relation between selected features and the delay, several illustrative plots were made.

Train interactions

Visual analysis of the impact of train interactions on the realized delay in the future provides an interesting observation. It is important to note that the relation between the features and delay can vary significantly among individual train series because every train series can face different amount of interactions of various importance. For the visual informativeness, the train series that provide well observable relations are selected. Understandably, depicting a well observable relation brings a certain bias in understanding the data. Nevertheless, we are aware that the role of each feature varies among the train series possibly from highly significant to negligible. And at this point, we are solely interested in the highest potential of the features.

In Figure 10 the various shades of grey of the markers in the plot depict different train series causing an interaction. Distribution of the delays shows that the longer delays do not necessarily occur where there is a large negative expected headway, which would suggest a higher probability of delay propagation. Instead, the higher delays can be observed rather when the expected headway is positive. However, an interesting phenomenon can be observed in the area of negative delay. There one can see that the train that is subject to prediction (Train series 700 as is the case of Figure 10) reaches negative delay (thus rides ahead of the schedule) in intervals of approximately 10 minutes, while in the middle of the intervals the delay raises towards zero. These areas are highlighted by the triangles inserted in the figures.

This phenomenon was observed with varying magnitude and intervals in a number of cases of the train series that are subject to the prediction. Moreover, this effect varies among the train series causing the interaction. Although it was possible to observe this effect also when the expected headway was positive, it can be assumed that the effect plays a more important role when there is a closer interaction and thus when the expected headway is negative. Assuming similar behavior of trains belonging to the same train type category (IC or SPR), plots were made showing the delay change from the current state to the delay 20 minutes in the future of trains belonging to the train series 700 with respect to the expected negative headway from trains of IC and SPR type. The plots can be seen in Figure 11 and Figure 12, respectively.

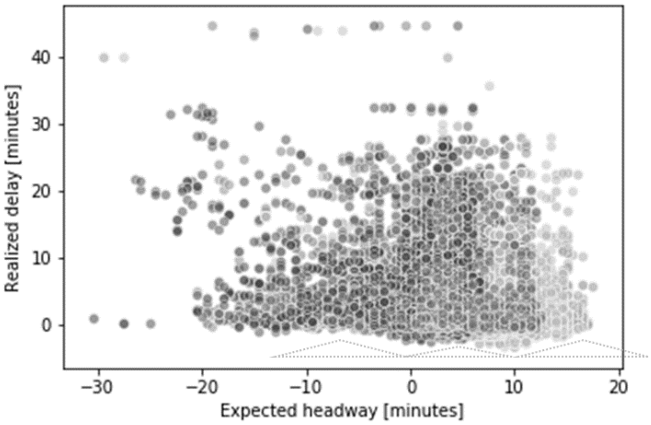


Figure 10 Expected headways to the interacting trains within the time window of 20 minutes and the realized delay 20 minutes in the future. Observed on train series 700 (SPR type).

In the case of an interaction with IC trains (Figure 11), one can see that there is a lower chance of a delay decrease when the expected headway is between approximately 2 and 10 minutes. The possible delay decrease drops from approximately 4 minutes to 2 minutes on the mentioned interval. Similar trend can be observed also on the interval shifted by 10 minutes towards larger negative expected headway. Train series 700 runs on the route Den Haag Centraal – Groningen with a frequency of one train connection per hour. The series interacts with 5 other train series belonging to the IC train type category and with 12 train series of the SPR train type. In the dataset of the train series 700, there are 175,092 instances. Within those, there are 5,164 identified interactions with a negative expected headway with IC trains and 16,183 interactions with SPR trains.

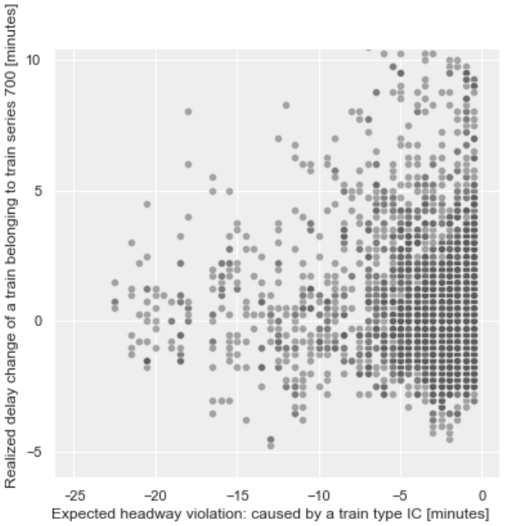


Figure 11 Expected violated headway between trains of train series 700 and relevant IC trains, and the delay of the trains of the train series 700.

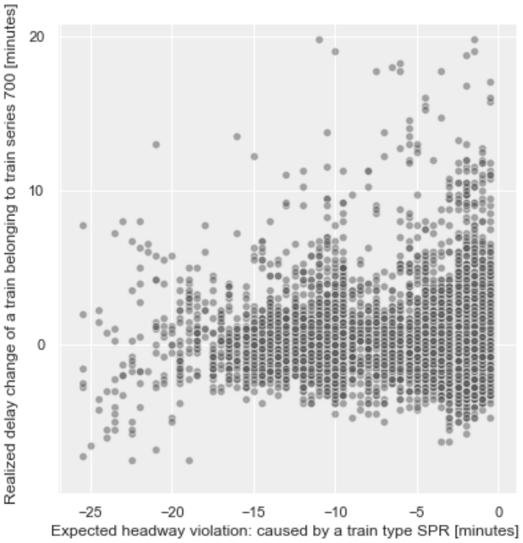


Figure 12 Expected violated headway between trains of train series 700 and relevant SPR trains, and the delay of the trains of the train series 700.

Looking at the specific trains series causing the interactions, the majority of the interactions occur with trains belonging to the train series 600 (between Zwolle and Meppel), 11600 (between WP and VTBR), 3300 and 4600 on the section between Leiden and Schiphol, and finally a vast majority of the interactions (almost 8,500 instances, thus almost a half of all the interactions) is with trains of the train series 14600 in the vicinity of Almere. And particularly the train series 14600 shows a very clear pattern in the limited delay reduction with the lowest delay reduction around the minimum required headways violated by 10 minutes.

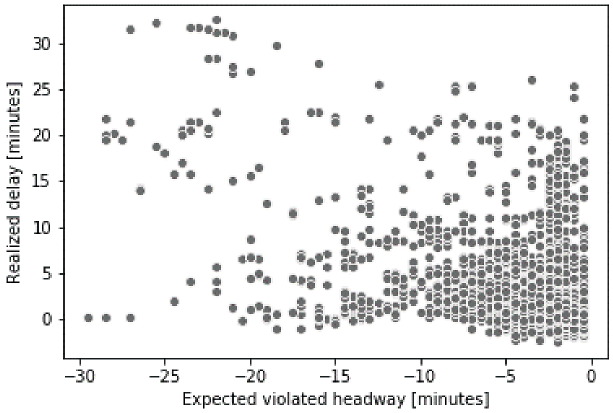


Figure 13 Expected violated headway between trains of train series 700 and trains belonging to the series 14600.

Obviously, the interactions are rather a sparse occurrence in the data. That combined with the delay distribution (as in Table 23) yields a rather low expected impact of the features on the predictions. However, there is a potential of the features having a positive impact on overly optimistic predictions forecasting decrease in the delay. Still, the number of observations falling to the category to which the observed tendencies apply is limited and so is the number of predictions that can be improved by these features.

Passengers

When assessing the relationship between the features and the future delay, it can be more informative to look at the delay change instead of the actual delay observed in the future because one of the features known to the model is the current delay. Beginning with the relationship between the seats ratio and the registered delay change (Figure 14), delay changes tend to be larger when the seating capacity is close to the originally planned capacity. Opposingly, the delay changes in both directions are lower when the realized capacity exceeded the planned capacity. The full range of observed delay changes occurred when the realized and planned capacity was equal, likely due to the fact that this group covers a vast majority of observations.

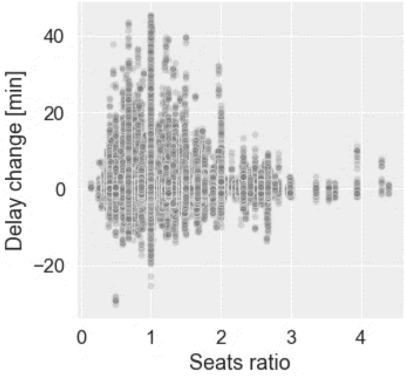


Figure 14 Ratio of the planned and realized seats and the delay change.

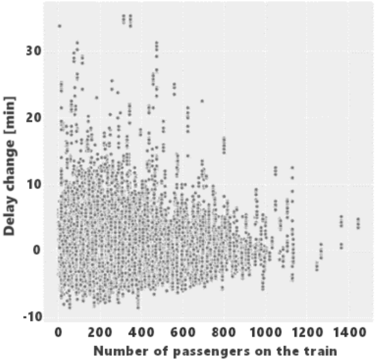
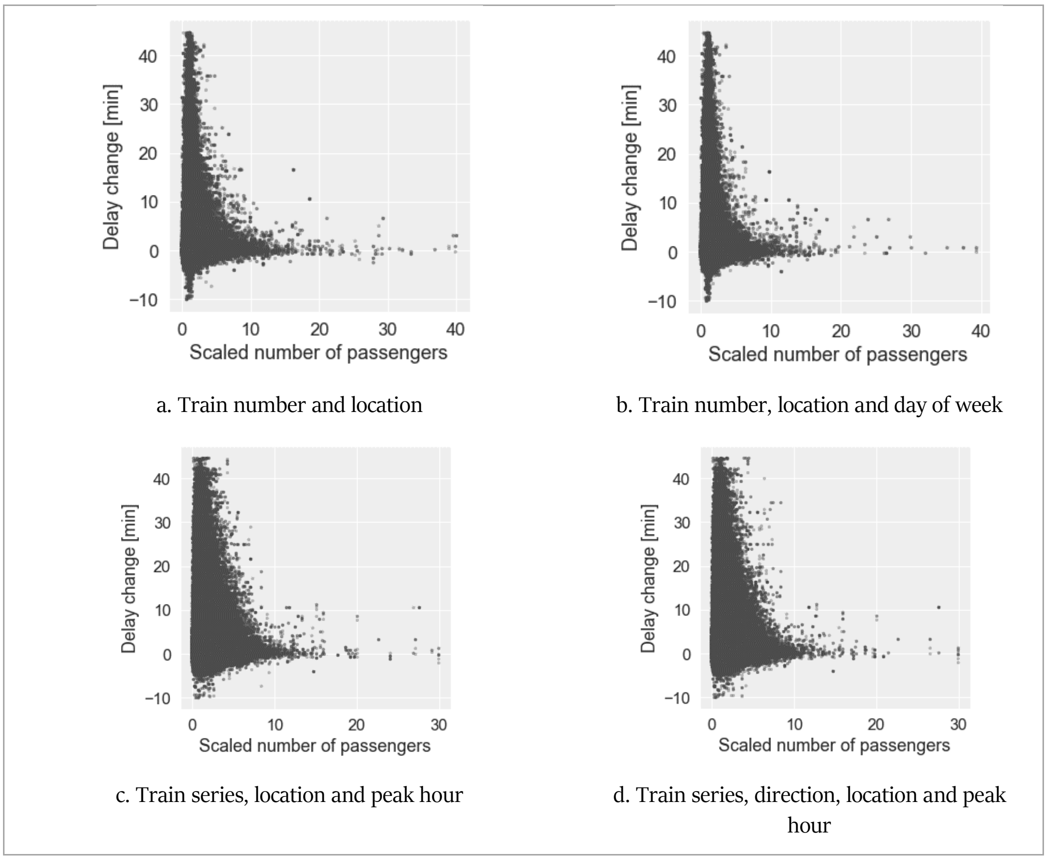


Figure 15 Number of passengers and the delay change.

Figure 15 then presents the relation between the number of passengers that departed from a station and delay change in 20 minutes from that point. Generally, the lower the number of passengers, the larger delay changes in both directions are observed. Nevertheless, the features reflecting the passenger data were created in 4 versions representing deviations of passenger volumes from a supposedly normal state for a certain location, time or train identification. The plots in Figure 16 show the scaled total number of passengers and the observed delay changes. The shapes visible in the individual figures vary slightly in the slope of the reduced variance in observed delay changes associated with the scaled numbers of passengers. Nevertheless, all of them reveal lower occurrence of a large delay increase when the scaled number of passengers increases. Slope of the maximum registered delay increase with an increasing number of passengers is the steepest in the sub-figure a. and gets gradually less steep towards to sub-figure d.. Similar tendency is observable in the delay decrease although with a lower magnitude of the delay change. Furthermore, a similar trend is visible in Figure 15, which depicts the relation between the delay change and the actual not-scaled number of passengers.



a. Train number and location

b. Train number, location and day of week

c. Train series, location and peak hour

d. Train series, direction, location and peak hour

Figure 16 Number of passengers departing from the last station prior to the current moment scaled to the median value per category (a. – d.) and delay change.

Weather

As can be seen in Figure 18 and Figure 20, high average wind speed above 10 m/s was measured along with a larger delay increase which applies also to the highest hourly wind speed. The average wind speed above 10 m/s was measured in 2,133,835 instances, above 11 m/s in 1,658,591 instances, above 12 m/s in 1,200,431 and above 13 m/s in 838,894. Even the last number of instances represents almost 10% of the instances and therefore there is a potential that the features representing wind speed will have an influence on the performance of the delay prediction.

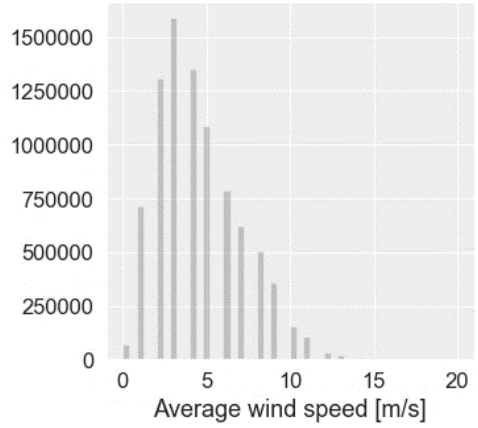


Figure 17 Distribution of average wind speed.

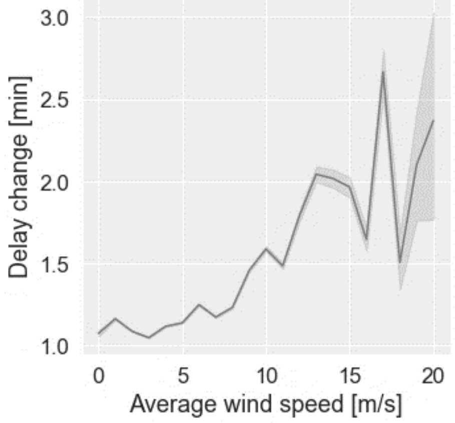


Figure 18 Average wind feature and corresponding delay change with 95% confidence interval.

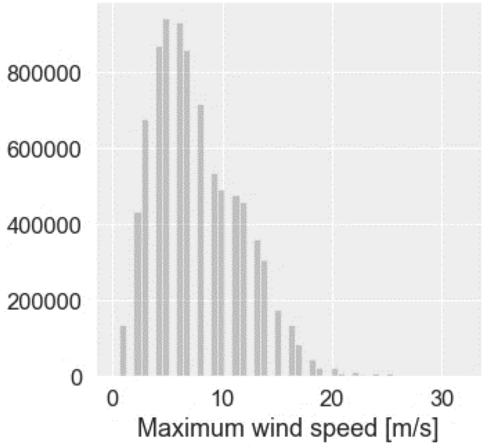


Figure 19 Distribution of maximum wind speed.

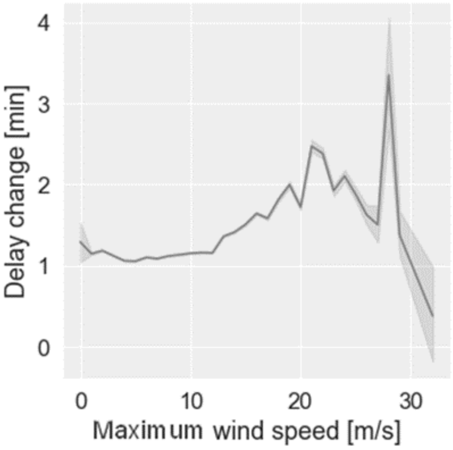


Figure 20 Maximum (High) wind speed and corresponding delay change with 95% confidence interval.

The observed delay change reaches the largest range in both directions when the lowest precipitation levels are measured, where also the absolute vast majority of observations belongs (see Figure 21). Figure 22 then reveals that most of the large delay changes are outliers beyond the 95% confidence interval (depicted as a shade around the line in Figure 22). There are large deviations in the mean observed delay change associated to a certain precipitation level. When very small amount of precipitation is observed, the delay change seems to slightly increase from where the tendency is rather declinatory. Clearer trend can be observed in Figure 24 showing the relation between temperature and the delay change. That is perhaps also due to wider distribution of the observed air temperature (Figure 23). In partial accordance with observations by Xia et al. (2013) presented in Section 2.4.2, low temperature is observed together with larger increase in delays while higher temperature is related to rather lower delay increase.

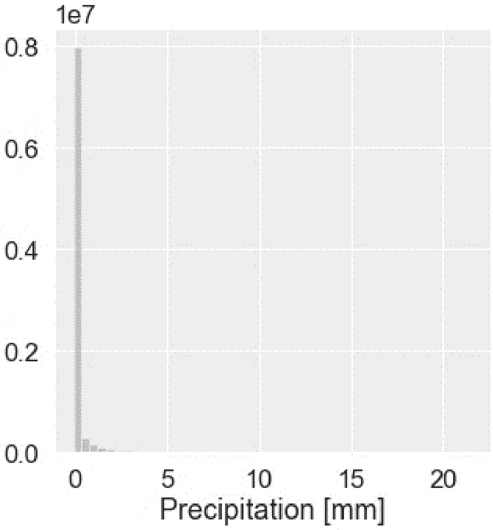


Figure 21 Distribution of measured precipitation.

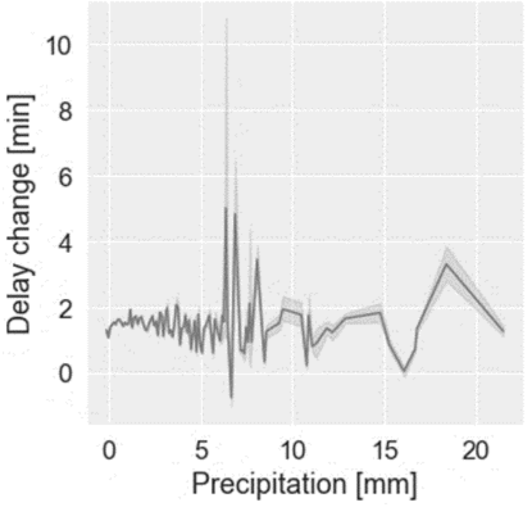


Figure 22 Observed precipitation and delay change.

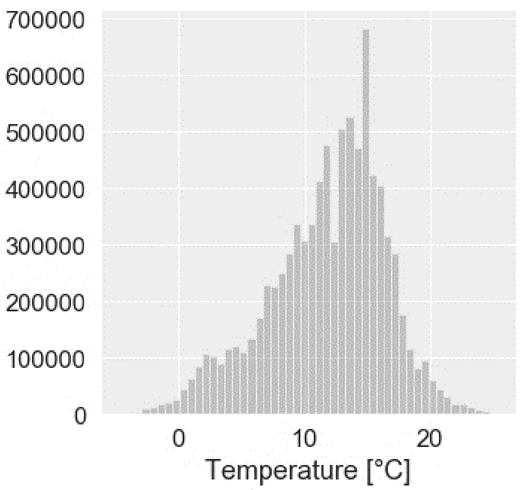


Figure 23 Distribution of measured temperature.

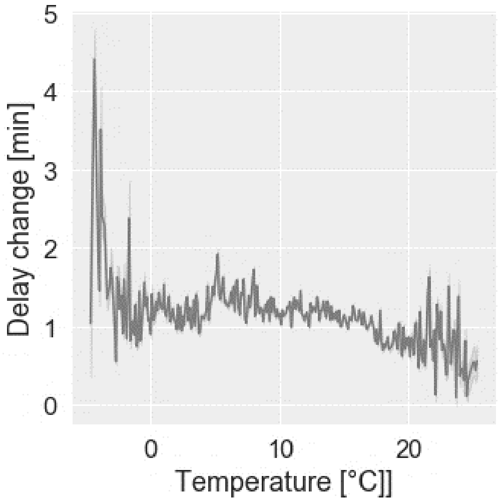


Figure 24 Observed air temperature and delay change with 95% confidence interval.

The binary weather condition features and the bad weather feature can be compared by the mean observed delay change. The overall mean delay change across the dataset is -0.025 with 95% confidence interval [-0.026, -0.024]. The mean delay change calculated per feature and its value is presented in Table 24 which reveals an increase in the mean value when any of the weather conditions are present. Considering the number of observations, especially 'Rain' appears to have a potential to have an impact in the delay prediction, while the remaining features will probably have only negligible impact.

Table 25 then shows an increase in the mean delay change, especially when 1 or 2 of the indicated weather conditions are present. A combination of 3 conditions shows a smaller increase in the mean delay change and appears only in an insignificant number of observations. The number of observations reporting one 'bad' weather condition present is close to the number of observations reporting presence of rain. Therefore there is a high probability that these two features are greatly similar.

Table 24 Binary weather features and corresponding delay change mean and 95% confidence interval (CI)

	False			**True**		
	Mean	95% CI	Number of observations	Mean	95% CI	Number of observations
Mist	-0.025	[-0.027, -0.024]	8,425,020	-0.020	[-0.026, -0.013]	301,044
Rain	-0.046	[-0.047, -0.045]	6,260,852	0.028	[0.026, 0.030]	2,465,212
Snow	-0.026	[-0.027, -0.024]	8,665,673	0.036	[0.022, 0.049]	60,391
Storm	-0.026	[-0.027, -0.024]	8,657,889	0.029	[0.016, 0.043]	68,175
Ice	-0.025	[-0.027, -0.024]	8,650,930	0.004	[-0.009 ,0.017]	75,134

Table 25 Bad weather feature and corresponding delay change mean and 95% confidence interval (CI)

	0		**1**		**2**		**3**	
	Mean	95% CI	Mean	95% CI	Mean	95% CI	Mean	95% CI
	-0.048	[-0.049, -0.046]	0.023	[0.021, 0.025]	0.025	[0.017, 0.034]	-0.004	[-0.032, 0.024]
Number of observations	5,948,623		2,590,867		170,975		11,979	

4.7. Summary

In this chapter, we presented and described the datasets we used in our work. The first one was the planned timetable, from which we retrieved features representing locations that a train will pass through in the delay prediction period of 20 minutes. Those features shall reflect busyness of the network the train will use in the 20 minutes, as a busy section may pose a higher risk of train interactions or more interactions with passengers. Next, the 'realization' data (AVL data) were introduced and the basic feature set (mainly temporal and spatial features and the present and past delay) derived together with the label (the future delay). Presentation of the newly introduced data followed afterwards along with introduction of the derived features reflecting train interactions, passenger counts and weather conditions. Finally, an overview of the data and a brief observation of the relation between the features and the future delay or delay change was presented. All the findings lead to answering the following research sub-questions.

Research sub-question(s) to be answered:

3. *How can/must be the data representing the factors modified to be used as features in the models expectedly maximizing their contribution to the model's performance?*
4. *Is there an observable relationship between the included factors and train delays?*

The train interactions data originating from the minimum required headway necessitate extensive data processing to retrieve the actual possible train interactions and conflicts. The planned headways on shared track had to be identified along with the minimum required headway which needed to be updated by the delays of the involved trains. The features were defined per train series to which the leading train belongs to and by a more concise category, a train type. That was based on the analysis of train interactions with respect to the future delay and delay change presented in Section 4.6, where we discovered patterns of a decreased probability of a delay reduction when the expected headway or expected violated headway reaches certain thresholds.

Although the passenger data could be directly used as a feature, scaling to reflect abnormalities rather than the total counts was opted for (as argued in Section 2.4.3). The selected scaling reflects various categories which may indicate the abnormalities such as train number or series, location, day of the week or direction. An observation of the data shows that there is a significant increase in delay length when the realized capacity is smaller than the planned capacity was. The relation between the delay change and the number of passengers yields a tendency of a decreased delay change in either direction with an increasing (or in a smaller extent also a decreasing) number of passengers. The scaled passenger counts then lead to a significant delineation of more sharply outlined patterns though with a very similar tendency.

The weather data did not require any transformation and could be directly used as features for the model. The only exception being the binary identified weather conditions, which were also transformed into a sum representing a presence of bad weather conditions with a higher severity when multiple weather conditions are present. That was done to replace the relatively sparse data of the binary identified weather conditions. Data analysis showed an increase in delay growth when high wind speed was measured. The delay change magnitude seems to decrease with an increasing level of precipitation or increased deviation of the temperature from the most frequently observed temperature. The binary identified weather conditions are identified as present rather sparsely except for the rain and mist. Presence of mist does not seem to have a significant impact on the delay change, but presence of rain suggests rather increase in delay as the mean delay change moves above zero. A similar effect is caused by the presence of snow and storm but the low number of observations where they are registered denotes a lower chance of having a significant impact on the results.

5. Model development

In this chapter, we summarize the feature sets developed in the previous chapter and present a scheme by which the individual feature sets and their versions are combined for the prediction models' development. Furthermore, we explain how the models are developed, and how the XGBoost hyperparameters are tuned. Eventually, a summary is given answering the relevant research sub-questions.

5.1. Feature sets: summary

A variety of feature sets built from numerous feature sub-sets and their versions was developed and tested in our research. It is therefore convenient to define terminology of the feature sets elements that is used hereinafter. The final feature sets that are used in the models consist of a number of feature sub-sets. Each feature sub-set contains features of a certain category (such as weather conditions) and may come in multiple versions. For an illustrative explanation of feature set composition, see Figure 25.

Five feature set categories were derived from the data. They will be referred to by the following names: '*Base*', '*Locations*', '*Passengers*', '*Weather*' and '*Interactions*'. Together with a recapitulation of the categories in the following text, the retrieved feature sub-sets and their versions are summarized. The value ranges and the variable types are presented along.

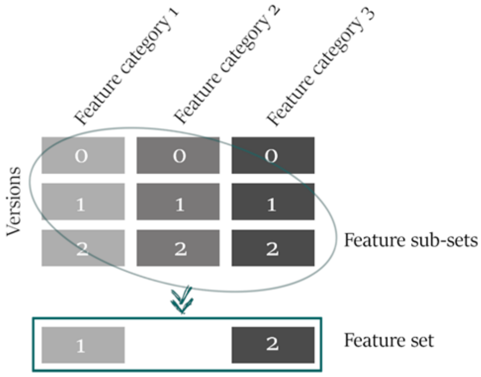


Figure 25 Feature categories, sub-sets and sets scheme.

The first feature category referred to as the '*Base*' contains basic spatiotemporal characteristics of the train events. In addition, the current delay and the delays registered at the last two registration points are in this feature sub-set. This feature set category has only one version and appears in all feature sets.

Base	0	Day of the week	[0,4]
		Hour	[0,23]
		Minute	[0,59]
		Location	[0,n] (n=number of locations in the network)
		Direction	[0;1]
		Current delay	ℝ (minutes with a precision of 0.25 min)
		Delay 1 before	ℝ (minutes with a precision of 0.25 min)
		Delay 2 before	ℝ (minutes with a precision of 0.25 min)

The second feature category derived from the timetable data is called '*Locations*' and is available in feature sub-sets of the two versions. Both versions represent 'busyness' of the trip section coming in the next 20 minutes. The first one contains the number of activities that are coming up, while the second version contains the number of occurrences of the upcoming locations in the timetable.

Locations	**0**	*Number of the activities to come:* V (departure) K_V (Short departure) D (pass through)	ℕ
	1	*Total frequency of the activities to come:* V (departure) K_V (Short departure) D (pass through)	

The next feature set category called '*Passengers*' evidently reflects passenger counts derived from fare collection data. However, the first version of this feature sub-set contains only the ratio of the realized and planned passenger seats, and an identification of a peak hour. The following versions 1 to 4 contain those two features complemented by the number of passengers boarding and alighting at and departing from the location. The passenger counts are scaled to a median value of a subset of the data sorted by varying factors depending on the feature set version.

Passengers	**0**	Seats ratio Peak departure	ℝ [0;1]	**3**	Seats ratio Peak departure Total p.* Boarding p.* Alighting p.* **Scaled by:* *Train series, location and peak hour*	ℝ [0;1] ℝ ℝ ℝ
	1	Seats ratio Peak departure Total p.* Boarding p.* Alighting p.* **Scaled by:* *Train number and location*	ℝ [0;1] ℝ ℝ ℝ			
	2	Seats ratio Peak departure Total p.* Boarding p.* Alighting p.* **Scaled by:* *Train number, location and day of the week*	ℝ [0;1] ℝ ℝ ℝ	**4**	Seats ratio Peak departure Total p.* Boarding p.* Alighting p.* **Scaled by:* *Train series, location, peak hour and direction*	ℝ [0;1] ℝ ℝ ℝ

The following feature set category called '*Weather*' represents various weather conditions including wind, temperature, precipitation and visibility. The second version contains a binary identification of presence of mist, snow, rain or storm in addition. The last version sums up all the binary variables from the second version and represents bad weather conditions.

Weather	**0**	Average wind speed Highest wind speed Temperature Precipitation View distance	ℕ ℕ ℕ ℕ ℕ	**2**	Average wind speed Highest wind speed Temperature Precipitation View distance Bad weather	ℕ ℕ ℕ ℕ ℕ [0,4]
	1	Average wind speed Highest wind speed Temperature Precipitation View distance Mist Snow Rain Storm Ice	ℕ ℕ ℕ ℕ ℕ [0;1] [0;1] [0;1] [0;1]			

Last, the feature set category representing train interactions referred to as '*Interactions*' has 6 versions. The first three versions contain as many features as there is train series the relevant train series (the one a prediction is made for) interacts with. First, value of the features is the expected headway, next of the expected violated headway and the third version is a binary identification of any existing interaction with the respective train series. The other three versions all consist of three features representing IC train, SPR trains and empty rolling stock. The expected headway and expected violated headway then refers to the first upcoming interaction with a train series belonging to the relevant category. The last version is a sum of the number of existing interactions within the relevant train type.

Interactions	**0**	Expected headway (per train series)	ℝ (minutes with a precision of 0.5 min)
	1	Expected violated headway (per train series)	ℝ (minutes with a precision of 0.5 min)
	2	Binary interaction identification (per train series)	[0;1]
	3	Expected headway (per train type)	ℝ (minutes with a precision of 0.5 min)
	4	Expected violated headway (per train type)	ℝ (minutes with a precision of 0.5 min)
	5	Binary interaction identification sum per train type	ℕ

5.2. Feature sets: A testing scheme

A number of combinations of feature sub-sets and their versions is compared and the best performing combination is to be selected. To do so in an orderly manner, a testing scheme was developed to define the order in which the models will be build. The order of the feature sub-sets combinations is presented in Table 26. The Base feature sub-set has only one version and the results are compared to the performance of the Naïve forecast and the results of Van den Bulk et al. (2018). Then, the two versions of Locations are added. The results are compared to the results from the previous step and between the two versions. The better performing Locations version is selected to be used along with the base for all the following steps. The Locations feature sub-set is added permanently to the Base sub-set as both sub-sets are retrieved from the original datasets (the Timetable and Realization data). On the other hand, the Base feature sub-set is based on the basic feature set of Van den Bulk et al. (2018) while the Locations sub-set is newly introduced. Thus, in sake of comparability and observability of the impact of the newly added features, the two sub-sets are first examined separately. Then, models using the Passengers, Weather and Interactions feature sub-sets are build and the results are compared the results of the relevant Base and Locations combination. Finally, the best performing version of each category is selected into the final model.

The models' performance is measured by the KPI's defined in Section 3.5.1. However, not all the KPI's are used when comparing the results in the testing scheme. The KPI's used in this phase are RMSE, MSE and the relevant confidence intervals for the regression task, and F1 score, precision and recall for the defined classification tasks. Eventually, the best performing feature sub-sets combination is analyzed more in depth, including the features importance to gain deeper understanding of the model's performance. Furthermore, the final model is again compared to the benchmark models' results.

Table 26 Testing scheme of combination of the feature sets and their variations

Step	Feature sets	Number of models
1	Base	1
2	Base, Locations	2
3	Base, Locations, Passengers	5
4	Base, Locations, Weather	3
5	Base, Locations, Interactions	6
7	Base, Locations, Passengers, Weather, Interactions	1

5.3. Model

The model described below is built to be used with already prepared features and labels. Therefore, the data processing and feature and label preparation are not part of the model. The model consists of multiple sections including the input data preparation, the prediction model tuning, the final prediction model development and the basic results assessment. Although an individual prediction model is built for each train series separately, the process of developing the prediction models for all the train series is integrated in the overall model. The details of the model are described in the following chapters.

5.3.1. Data splitting

The dataset was split into three subsets: train ('train'), validation ('valid') and test ('test') set. The data splitting into the subsets was done for each train series individually by randomly selecting subsets of train trips. There were between 26 and 4,572, and on average 2,825 train trips registered for the train series. Each train trip was then registered at 3 to 77 registration points with an average of 28 registration points. Each registration then is an individual instance where the features and label values are taken from. The split to the set was done so that the train set consisted of 60% of the train trips and the validation and test sets of 20% of the train trips each. By doing so, the train sets contained between 16 and 2,743 train trips with an average of 1,695 trips. Only 4 train series were represented by less than 100 train trips in the training set. Accordingly, 5 to 914 train trips were assigned to the validation and test sets.

The split of the data on the train trips was done to avoid overfitting as it is likely that registrations within each train trip are highly correlated (e.g. weather conditions were based on hourly measurement and passenger counts were identical for all 'D' (pass through) events). Another option for the split was to do so based on time such as days. As the data consists of 14 weeks, only working days are considered and above that all Wednesdays are discarded due to irregular operations in the observed period, only 56 days are available. Therefore, 60% of the days would mean that only 34 days would be in the training set, which was concluded that it could have an impact on the representativeness of the data with respect to possible states of the system.

5.3.2. Structure of the model

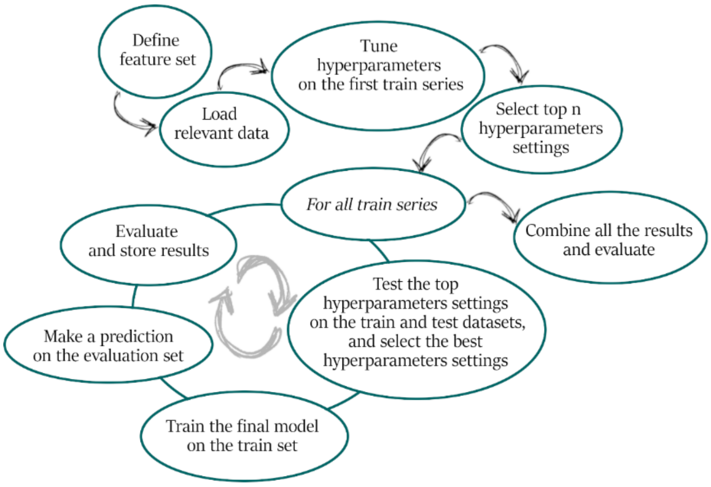

Define feature set
Load relevant data
Tune hyperparameters on the first train series
Select top n hyperparameters settings
For all train series
Combine all the results and evaluate
Evaluate and store results
Test the top hyperparameters settings on the train and test datasets, and select the best hyperparameters settings
Make a prediction on the evaluation set
Train the final model on the train set

Figure 26 Outline of the model's structure including hyperparameters tuning, and training, testing and evaluation of the prediction models developed per train series.

The model, of which the simplified structure is visualized in Figure 26, begins with a defining a set of variables that are used later on. First, the feature sets and their versions to be used are defined. Then, Genetic Algorithm (GA) parameters and stopping criteria for XGBoost hyperparameters tuning are defined. The GA parameters are the number of parents (population size; *n_parents*), the number of parents mating (*n_parentsmating*) and the maximum number of generations to be made (*n_generations*). The maximum number of generations is not reached if the stopping criteria are met. Those were defined such that the change in the best objective value across the population must be smaller than a defined threshold (*Min_Change*) for a minimum number of generations (*Min_Change_rounds*).

Table 27 Variables values used in our model.

Variable	Value
n_parents	45
n_parentsmating	30
n_generations	30
Min_Change	0.0001
Min_Change_rounds	5
top_n	15

The GA parameters were selected based on recommendations by Cox (2005) and Whitley (1994), and can be found in Table 27. The stopping criterium was defined considering recommendations by Bhandari, Murthy, & Pal (2012) who emphasized the risk of premature stopping when the stopping criterium is solely based on the objective's improvement over a set number of generations. Empirical observations of the objective value development showed that the objective is either relatively high or fluctuates around a stable relatively low value with only minor changes.

The possible values of the XGBoost hyperparameters (presented in Table 28) are defined (*Param_Values*). The values were based on and extended on those used by Van den Bulk et al. (2018) while considering what values performed the best in their work.

Table 28 XGBoost hyperparameters available for tuning.

Parameter	Possible values
learning rate	[0.001; 0.01; 0.1]
n estimator	[500; 750; 1000; 1250]
max depth	[5; 6; 7; 8; 9; 10]
child weight	[1; 2; 3; 4; 5]
gamma	[0; 0.1; 0.2; 0.3; 0.4]
alpha	[0; 0.01; 0.1]
subsample	0.8
colsample_bytree	0.8

Due to the size of the problem of the hyperparameter tuning, and the time necessary to train and evaluate the XGBoost model built for every combination of the hyperparameters, the tuning using the GA was done only for the first train series of each feature set. It is assumed that the optimal values are likely to be similar among the train series as the same features are used. However, especially due to the varying size of the data available for each train series, the optimal hyperparameters still might vary. To accommodate this expected variation, while avoiding an excessive runtime of the algorithm, n (*top_n*) best performing hyperparameters sets from the GA are selected, which become candidate sets for the rest of the train series.

Finally, the data, the label and the set of train series to build the prediction model for are loaded. In sake of comparability of the results, the split of the data into the train, validation and test set was done beforehand so that the subsets contain the same trips across the varying feature sub-sets combinations.

The algorithm of the model is presented in the following pseudocode (Figure 27). In a brief summary, the variables are defined and described input loaded. The model iterates over all the train series and therefore selects data relevant for the respective train series. The data is split to the training, validation and test set. In the first iteration (the first train series), the XGBoost hyperparameters are tuned using the GA.

Structure of the main model

```
Start
Define feature sets versions
Define GA parameters
Define stopping criteria
Define possible values of XGBoost parameters as Param_Values
Define number of top_n Param_Combs
Load data
Load label
Load Train_Series
Load splitting identification
Do_hyperparameters_tuning = True
FOR s in Train_Series
    X = data[s]
    y = label[s]
    Remove empty columns in X
    Split X to [x_train, x_valid, x_test] according to the splitting identification
    Split Y to [y_train, y_valid, y_test] according to the splitting identification
    IF Do_hyperparameters_tuning is True:
        Optimize XGBoost hyperparameters on [[x_train, y_train], [x_valid, y_valid]]
        Store top_n hyperparameters combinations in Param_Combs
        Do_hyperparameters_tuning = False
    END IF
    IF Do_hyperparameters_tuning is False:
        FOR p in Param_Combs:
            Train XGBoost on [x_train, y_train] with hyperparameters p
            Make a prediction on [x_valid, y_valid]
            Compute and store MSE of the prediction
        END FOR
        Select p from Param_Combs corresponding with the lowest MSE
    END IF
    Train XGBoost on [[x_train, y_train]] with hyperparameters p
    Make a prediction on [x_test, y_test]
    Analyze and store results
END FOR
Analyze and store overall results
End
```

Figure 27 Pseudocode of the main model.

The GA is terminated once the stopping criterium is reached or when the maximum number of generations is reached. *Top_n* hyperparameters combinations are selected (*n* ‘parents’ with the lowest objective value) combinations are stored. The rest of the algorithm applies for all the series again. An XGBoost model is built for the *top_n* hyperparameters combinations and the one with the lowest objective value is selected as the final hyperparameters setting. Finally, an XGBoost model is trained on the training set and eventually used to make a prediction on the test set. The MSE is calculated and the predictions and results are stored. When the iteration process over the train series is finalized, the overall MSE is computed and stored along with the predictions.

Structure of the XGBoost hyperparameters' optimization model applying Genetic Algorithm:

```
Input data:
  GA parameters:
    - n_parents
    - n_parentsmating
    - n_generations
  Stopping criteria:
    - Min_Change
    - Min_Change_rounds
  Param_Values
  x_train, y_train
  x_test, y_test

Start
Load input data
Create storage for Fitness_history
Create storage for Population_history
Create storage for Best_Fitness_Per_Gen
Generate random initial population
stop=0
i=0
WHILE i < n_generations
    j=0
    IF j < n_parents
        Train XGBoost on [x_train, y_train] with hyperparameters = population[j,:]
        Test XGBoost on [x_valid, y_valid] with hyperparameters = population[j,:]
        Compute fitness
        Store fitness in Fitness_history
        j=j+1
    END IF
    Store best fitness of the population in Best_Fitness_Per_Gen
    IF (i > 0) & ( Best_Fitness_Per_Gen[i-1] - Best_Fitness_Per_Gen[i] < Min_Change):
            stop = stop + 1
    END IF
    IF stop == Min_Change_rounds:
        BREAK WHILE
    END IF
    Select n_parentsmating parents with best fitness to mate
    Perform crossover to create (n_parents - n_parentsmating) children
    Mutate children
    Combine parents and children to population
    Store population in Population_history
    i=i+1
END WHILE
End
```

Figure 28 Pseudocode of the XGBoost hyperparameters' tuning.

The XGBoost hyperparameters tuning using a GA is described by the pseudocode above (Figure 28). The input data are loaded, and storage variables are created. The initial population ('parents') is created by random selections from the Param_Values. While the maximum number of generations has not been reached, each parent is used to train an XGBoost model on the train set. The model is used for a prediction on the validation set. The fitness is calculated as MSE of the prediction. The results are stored. The best fitness of the generation is stored. If the best fitness has not changed by *Min_Change* or more for *Min_Change_rounds*, the while loop is terminated. Otherwise, the fittest parents are selected for mating. Mating is done by a uniform crossover between pairs of parents. The children are mutated on one randomly selected gene. The population is updated and stored, and the next iteration begins.

XGBoost configuration

Besides the hyperparameters that are tuned in the main algorithm, other parameters defining characteristics of the models built by XGBoost have to be specified. In the general parameters, '*gbtree'* (a tree-based model) was selected as a booster. Next, in the task parameters, the objective was set to '*reg:squarederror*' (regression: squared error) and '*rmse*' was selected at an evaluation metric. The configuration is based on what was used by Van den Bulk et al. (2018).

5.4. Summary

There were two main goals of this chapter: to define feature sets and to present a design of the model. The goals were defined in the form of research sub-questions as follows.

Research sub-question(s) to be answered:

5. *What are sound feature sets?*
6. *What is a sound model structure including parameter tuning?*

Based on the literature study combined with expert opinions of Fioole & Tielman (2019), a set of factors that may influence development of delay in passenger railway was selected: passenger volumes, weather conditions and interactions among trains. In Chapter 4, the relevant data were described, and features were derived. In this chapter, we presented an overview of all the features along with a scheme by which possible combinations of the features aggregated in feature sub-sets were tested. All of the feature sets contain the '*Base*' which is then accompanied by the '*Locations*'. Those features are all timetable based complemented by the registered delays. Next, '*Passengers*' and '*Weather*' are added, first each separately, later together. These features are based on historical data. Finally, '*Interactions*' are added, first to '*Base*' and '*Locations*' only, and finally to the fully developed feature set also including the best performing combination of '*Passengers*' and '*Weather*'. The 'Interactions' add another dimension of timetable-based features updated by the present state of the system. It is important to keep in mind that the features and the feature sub-sets are a small instance of all the possible ways the features may be defined and combined. Nevertheless, potential of the features in the delay prediction in the context of our research will be uncovered in the following Chapter 6 Research results.

To answer the research sub-question 6, an algorithm was developed to build a model for each train series separately. As the hyperparameters optimization is a time-consuming step, it is performed only once for each feature set. A set of the best performing hyperparameters combinations is then used as a set of candidates for the remaining train series. The hyperparameters tuning is done using a training and validation set. The final model is trained on the train set and a prediction is made on the test set. All the predictions along with the true values are stored for an overall evaluation of the feature set across all the train series. Assessment of the models' performance within the tuning and training algorithm is done based on MSE while other KPI's are used in assessment of the prediction quality.

6. Research results

In the previous chapters, we presented the applied methodology from a theoretical perspective, then we proceeded to the input data description and consequently to outlining the model we developed. In this chapter, we present the results of the models' application. In total, 18 feature sets were used to build models for delay prediction of trains operated within 79 recognized train series. Those included all the feature sets categories as introduced in the earlier chapters. Due to computational power limitations, the largest feature sets reflecting train interactions were tested on a sample of train series, specifically on a set of 27 train series which were the first 27 train series in a numerical order of the train series numbering. Size of the subset results from the technical possibilities to ensure feasibility of the models' development. Results of these models are presented and analyzed apart as they cannot be compared to the results including all the train series.

First, results of the hyperparameters tuning are shortly presented. The rest of the chapter is focused on the prediction results. The results are presented by the key performance indicators presented in Section 3.5. Therefore, the models' performance is compared by the RMSE and related metrics, by the scores resulting from confusion matrices, and eventually, features' importance is inspected. The results are presented on the feature set level (see Figure 5 in Section 3.5). As the performance of the models built on the diverse feature sets considerably varies among the train series, results of the individual train series are presented in an appropriate form when relevant. Furthermore, the best performing model with respect to the RMSE was selected for each train series, the resulting predictions were combined across all the train series and evaluated as if each train series was predicted with the best fitting feature set. In addition, some results are visualized.

Note that the feature set categories will be identified by following abbreviations**: B (Base), L (Locations), P (Passengers), W (Weather) and I (Interactions)**. Versions of the feature sub-sets within the feature categories are identified by a number as is used in Section 5.1. As an example, a feature set B0L1W2 consists of the 'Base' feature sub-set, 'Locations' feature subset version 1 and 'Weather' feature sub-set version 2.

Naïve forecast: The reference

The Naïve prediction was computed as strictly subtracting 1 minute from the 'current' delay. However, the prediction by the model was done to the last train registration within a 20 minutes time window since the last registration point. Therefore, the exact prediction period might not be, and likely is not, precisely 20 minutes but a little less. To be truly exact, one should perform the Naïve forecast with a corresponding delay reduction which would then be a little less than 1 minute. However, the difference would be in a magnitude of seconds and unless it would be more than 15s (precision of applied rounding: 0.25 minute) corresponding with a prediction time window of 15 minutes, it should not affect the results significantly. Therefore, for simplification, the delay reduction is always assumed to be 1 minute.

6.1. XGBoost tuning

Adequately selected hyperparameters of the XGBoost model are a crucial element in the models' quality. The parameters were tuned using a genetic algorithm on the models developed for the first in order train series, and were used for all feature sets consisting of sub-sets from the relevant feature set categories. In the first generation, a completely random population counting 45 individuals (sets of the hyperparameters) was drawn from the allowed values of the hyperparameters. The hyperparameters setting was assessed by a MSE of the resulting prediction. An example of the obtained MSE's of the 45 individuals per generation are demonstrated in Figure 29. Note that each generation counts 45 individuals and the reduced number of points in the figure implies that the MSE's are close or nearly identical.

Figure 29 shows the obtained MSE's when only the 'Base' feature set category was used. It is possible to see that the individuals' performance varies largely in the first four generations only. In each generation, one parameter mutated. In the generation 9, apparently a more sensitive parameter was subject to mutation and several individuals with a worsen performance were created. Nevertheless, the stopping criteria were met at the end of the 12th generation.

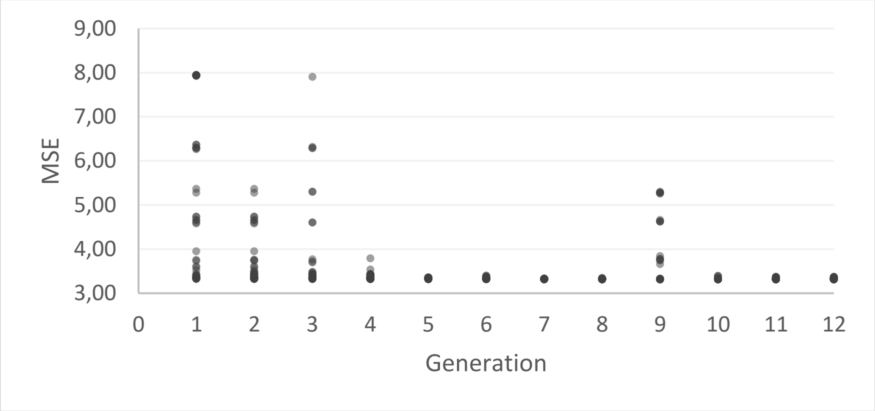


Figure 29 XGBoost hyperparameters tuning example. MSE values of 45 individuals across 12 generations.

For each combination of the feature set categories, a cluster of top 15 candidate hyperparameters arrays was selected. The algorithm did not restrict presence of duplicate candidates and therefore, between 7 and 13 candidates were available at the end. The candidates obviously varied among the feature set categories; however, the later selected candidates were rather uniform among the train series. The allowed values for each hyperparameter can be found in Table 28 in Section 5.3.2. All the candidates across all the feature set combinations favored a *learning rate* of 0.01 and no other value appeared in the candidates. The value of *n-estimators* varied between 750, 1000 and 1250, and was never 500. *Max depth* appeared in all of the possible values; however, it was 8, 9 or 10 in the most cases. Interestingly, *max depth* equal to 5 was used by the very most models in the feature sets including train-type-based train interactions. *Child weight* also appeared in all the possible values, and so did *gamma* and *alpha*. The candidates, that were selected for models of the most train series within each feature set category, are presented in Table 29 (the relevant feature set categories are depicted by the filled cell). Note that only train-type-based train interactions were considered in the data in the table, as the feature sets containing train interactions by train series were not applied to all the train series.

Table 29 XGBoost hyperparameters candidate sets performing the best with the most train series.

B	L	P	W	I	learning rate	n estimator	max depth	child weight	gamma	alpha	Selected for % of series
■					0.01	1250	7	5	0.3	0.00	68%
■	■				0.01	1250	6	5	0.3	0.00	42%
■	■				0.01	1250	6	5	0.0	0.10	48%
■	■	■			0.01	750	8	2	0.4	0.00	18% – 38%
■	■		■		0.01	750	9	4	0.3	0.01	41% - 52%
■	■		■		0.01	750	9	3	0.3	0.10	30% - 38%
■	■			■	0.01	1250	5	5	0.4	0.00	70% - 80%
■	■	■	■		0.01	1000	8	4	0.2	0.10	22% - 30%
■	■	■	■		0.01	1000	8	4	0.4	0.10	19% - 28%

6.2. Regression

Basic assessment of the regression results is done by comparison of the RMSE's. All the results are presented in Table 30 along with the corresponding MSE's and the 95% confidence intervals. Next, the number of train series that reach the lowest RMSE using the respective feature set is presented, the number of train series outperforming a reference feature set, and finally a number of train series performing the best within the relevant feature set category, when the respective feature set is used. Focusing on the RMSE, results of the B0 feature set correspond with our expectations, as the RMSE of 1.52 outperforms the Naïve forecast's RMSE of 1.99 but not the results of the best performing model from the research by Van den Bulk et al. (2018), which reached a RMSE of 1.34 and which included further features in comparison to our basic feature set. Unfortunately, the form of the available results of Van den Bulk et al. (2018) does not provide sufficient information to perform more detailed analysis. The lowest overall value (RMSE 1.48) is reached with the feature sets that include train interactions either in the form of expected headway (B0L1I3) or expected violated headway (B0L1I4). The latter one shows a higher upper bound of the confidence interval by 0.01. Nevertheless, both reach the lowest RMSE by the most train series (29% and 16% respectively). The remaining train series reach the best performance with feature set B0L1W1 (10% of train series), B0L1 (9% of train series) followed by B0L1W2, B0L1I5 (both 6% of train series) and B0L1W0 (5% train series). Within the subset of 27 train series that were used for comparison of the feature sets containing all versions of train interactions feature sub-sets, the feature sets containing features per train type surprisingly outperformed their more complex alternatives in both, the overall RMSE and the number of train series performing the best when the respective feature set is used.

For further comparison of the feature sets by performance of the predictions on the train series level, we calculated the number of train series that reach lower RMSE compared to a reference feature set. Two reference feature sets were defined: B0 as the very basic feature set and B0L1 as the enlarged basic feature set included in all the later feature sets. Next to that, we calculated the number of train series for which the relevant feature set outperforms other feature sets within the same category. Note that the comparison was done based on RMSE rounded to three decimal points and thus the results associated with a certain train series could show equal RMSE's of multiple feature sets. The rounding was done to neglect insignificant differences among the RMSE's. Both versions of the Locations features decrease the RMSE for majority of train series compared to the results of B0. Comparing the two versions, there is a significantly larger number of train series that reach a better result with B0L1 (58 train series) than B0L0 (30 train series). This supports our selection of B0L1 to be included in the later feature sets instead of B0L0. Next, looking at the feature sets including Passengers features, the number of observed improvements among the train series compared to the B0 is significantly lower than is the case of B0L1, which suggests that for many train series, inclusion of the Passengers features brings rather worsening of the results than an improvement. Compared to the second reference feature set (B0L1), only 10-17 train series out of 79 reach lower RMSE when the passengers features are included. Nevertheless, comparing the versions of the Passengers feature category, the simplest one B0L1P0 performs the best for approximately 40% of the train series. Very comparable results are observable when the Weather features are included, although there are more train series for which the prediction improved when the Weather category was included compared to the reference B0L1. And again, the simplest version B0L1W0 outperforms the other versions by 44% of the train series, reaching its lowest RMSE compared to B0L1W1 and B0L1W2. Finally, the Interactions features bring an improvement to a majority of train series compared to both references. Thus, the features do not deteriorate the predictions quality as Weather and Passengers do in some cases. Specifically, predictions of 67% - 89% of the train series gain an improvement when Interactions are included compared to the B0L1 feature set. Unlike was the case of Weather and Passenger feature categories, many train series reach equal results among the Interactions feature sub-set versions. 13 train series show the same RMSE of the predictions using either of the three versions. 11 train series out of those do not reach their overall best result when the Interactions are included, and their results are comparable to the corresponding results of B0L1. In those cases, it can be assumed that the

Interactions features simply do not have any impact no matter the formulation. In the case of the two train series when all the three feature set versions show nearly equal RMSE (equal when rounded to 3 decimal points) while some of them is the overall top-performing feature set of the train series, the Interactions features apparently represent the interactions in a highly comparable way, possibly referring to the same interactions.

Next to the generic RMSE, a RWMSE was defined to emphasize errors associated with large observed delays. See the definition in (3.5) and (3.6), Section 3.5. The RWMSE was calculated on the feature set level only. Assessing the feature sets based on this metrics, the differences become significantly smaller. The best performance is observed on the feature sets B0L1, B0L1I3-5 with RWMSE of 1.35, followed by B0L0 and B0L1P0 with RWMSE of 1.36. The remaining feature sets reach RWMSE of 1.37 with an exception of the B0 feature set that is burdened by a RWMSE of 1.38.

Based on the RMSE and the RWMSE, the best performing versions of the feature set categories were selected. For the Passenger category, it was version 0 and so it was for the Weather category. In selection of the best performing version of the feature set containing the Interactions category, the number of best performing train series was considered in addition to the RMSE and the RWMSE. Thus, version 3 was selected. Selection of the versions is in line with performance of the individual train series within the feature categories, as the most train series reach their best results in the categories with the sub-set versions L1, P0 and W0 respectively. Altogether, the final feature set can be denoted as B0L1P0W0I3. This feature set was created despite the poor performance of some of the feature sub-sets in the simpler feature sets. The purpose of this feature set was to observe whether the feature sub-sets combined together lead to an improvement or worsening or no change in the performance. As can be observed, the RMSE and RWMSE are both worse than was the case for B0L1 and B0L1I3-5. On the other hand, the RMSE of 12 train series (15%) reached its minimum with this feature set. 1 IC train series originally performed better with the feature set B0L1W0. The other train series belong to the SPR type and originally performed the best with the following feature sets (number of relevant train series): B0L1P1 (1), B0L1W0 (2), B0L1W1 (3), B0L1W2 (3), B0L1I3 (2). Compared to the reference feature sets, the complex feature set brings an improvement to a significantly lower number of train series than feature sets of the Interactions category did. Therefore, the complex feature set does improve the prediction for a few train series, however, many train series benefit more from the simpler feature sets B0L1I3-5.

Finally, see the lower part of Table 30 for comparison of all the six feature set versions including Interactions features. Note that the prediction was made for a subset of 27 train series and the results cannot be directly compared to the results in the upper part. Instead, the results provide a comparison among the versions of the train interactions feature sets. Contrary to our expectations, looking at the RMSE, the overall results are slightly worse when the interactions are defined per individual train series (versions 0-2) instead of train types (versions 3-5). That can be seen also in the number of train series that reach its best predictions when the relevant feature sets are used. Out of the set of 27 train series used to test all the Interactions feature sets, 14 train series reached the best performance when the interactions features were represented per train type while 6 train series reached the lowest RMSE with features representing interactions with individual train series. On the other hand, the difference between the feature sets diminishes when the RWMSE is used as an evaluation measure. Nevertheless, the binary encoded Interactions features in the B0L1I2, and the corresponding simplified version B0L1I5 are still clearly the least effective.

Table 30 RMSE, MSE, 95% Confidence intervals of MSE and RWMSE of the regression task per feature set, and the number of train series reaching the lowest RMSE with the respective feature set: overall, compared to a reference feature set and within a feature set category.

	B	L	P	W	I	RMSE	MSE	95% CI	RWMSE	Best in series	Best in SPR series	Best in IC series	Series outperforming reference feature set		Best in series per feature set category
	Naïve					1.99	3.94	[3.90, 3.98]	1.91						
	0					1.52	2.30	[2.27, 2.33]	1.38	2	0	2	reference	-	
	0	0				1.50	2.24	[2.21, 2.27]	1.36	2	2	0	73	-	30
	0	1				1.49	2.23	[2.20, 2.27]	1.35	7	5	2	73	reference	58
	0	1	0			1.50	2.26	[2.23, 2.29]	1.36	2	1	1	54	10	31
	0	1	1			1.50	2.26	[2.23, 2.29]	1.37	3	2	1	45	14	11
	0	1	2			1.51	2.27	[2.24, 2.30]	1.37	1	1	0	47	15	12
	0	1	3			1.51	2.27	[2.23, 2.31]	1.37	3	1	2	47	16	10
	0	1	4			1.51	2.27	[2.24, 2.30]	1.37	1	0	1	43	17	18
	0	1		0		1.50	2.26	[2.23, 2.30]	1.37	4	2	2	47	23	35
	0	1		1		1.51	2.27	[2.24, 2.30]	1.37	8	7	1	39	22	21
	0	1		2		1.51	2.27	[2.23, 2.30]	1.37	5	4	1	40	23	23
	0	1			3	1.48	2.20	[2.17, 2.23]	1.35	23	11	12	74	60	47
	0	1			4	1.48	2.20	[2.17, 2.24]	1.35	13	6	7	75	59	34
	0	1			5	1.49	2.22	[2.19, 2.26]	1.35	5	4	1	74	53	25
	0	1	0	0	3	1.50	2.25	[2.21, 2.28]	1.36	12	11	1	52	35	
Results of a subset of train series	0	1			0	1.56	2.43	[2.38, 2.47]	1.41	3					
	0	1			1	1.56	2.43	[2.38, 2.47]	1.41	2					
	0	1			2	1.57	2.45	[2.40, 2.49]	1.42	1					
	0	1			3	1.55	2.41	[2.37, 2.46]	1.41	7					
	0	1			4	1.55	2.41	[2.37, 2.46]	1.41	6					
	0	1			5	1.56	2.44	[2.40, 2.49]	1.42	1					
	All combined					1.48	2.20	[2.16, 2.23]	1.34						

All the RMSE's of all the models on the train series level can be found in Appendix B.1, Table 52 (results of IC train series) and Table 53 (results of SPR train series). The values are highlighted by colors representing the order from the highest to the lowest RMSE per train series. The very lowest RMSE of each train series is further highlighted by an outline of the cell. Looking at the results in the mentioned tables, one can see that there are mostly very minor differences among the versions of the feature set categories. Some differences are not observable as the RMSE is displayed with two decimal points precision. The variations in the features definition therefore appears to have a very limited impact, if any, in many cases. For visual overview of the results on this level, see Figure 30 depicting the order of the RMSE's belonging to each train series.

In summary, assessing the models based on the RMSE and the related metrics, the best performing models are generally those that include features representing the upcoming activities and locations ('Locations'), and the train interactions features. The passengers features all seem to perform comparably and rather introduce worsening of the results when introduced. So do weather conditions features and their combinations. Nevertheless, performance of the feature sets highly varies among the train series and all of the feature sets perform the best for at least one of the train series. Therefore, we selected the best model for each train series and assessed the overall result. The resulting RMSE is equal to the lowest RMSE that was associated with the feature sets B0L1I3 and B0L1I4. However, the confidence interval of the MSE is slightly shifted as the lower bound decreased, and the RWMSE reaches the overall minimum.

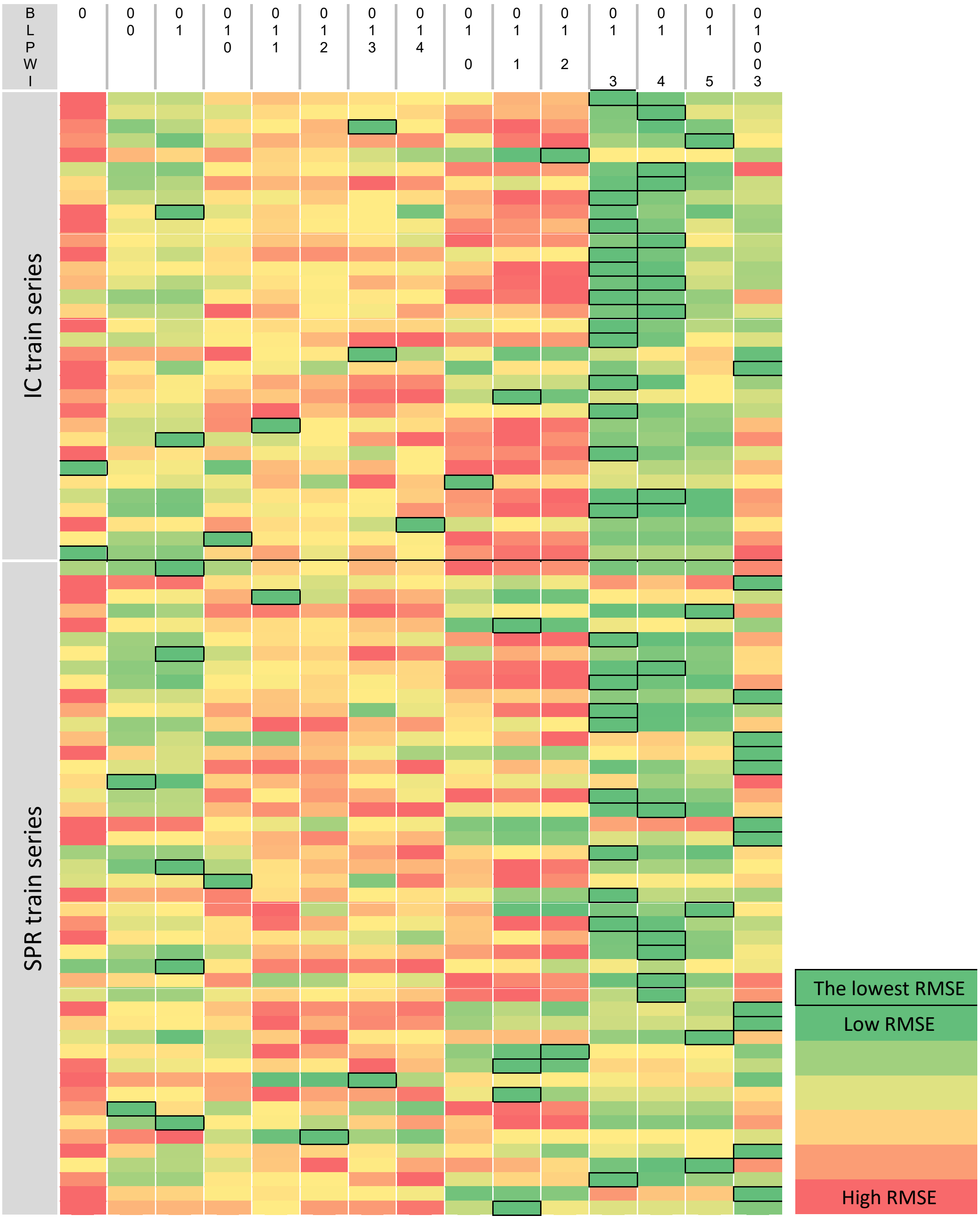


Figure 30 Visual representation of delay prediction performance comparing the RMSE of various feature sets for all the train series.(Feature sub-sets abbreviations: B (Base), L (Locations), P (Passengers), W (Weather) and I (Interactions))

6.3. Discretized prediction

To assess the predictions' performance with respect to the delay length, the measured and predicted delay was rounded to full minutes. Negative delays were set together with a delay of 0 minutes as a train running ahead of the schedule is assumed to be preventable. Delays above 15 minutes were put in the bin of 15 minutes delay that marks a threshold for 'large' delays requiring greater interventions. Finally, a confusion matrix was made and the resulting precision, recall and F1 score was calculated per delay length and for each feature sets combination.

The resulting F1 scores are presented in Figure 31, where the Naïve forecast, the B0 feature set and the 'all combined' set's results are highlighted by differing marks. All the feature sets perform significantly better and comparably in prediction of no-delay and small delays. The difference from the Naïve forecast gradually decreases towards a delay of 6 minutes from where the Naïve forecast begins to outperform our models and from a delay of 9 minutes onwards, the Naïve forecast outperforms all the other models. Predictions of delays equal to and exceeding 15 minutes are then again reaching a high F1 score thanks to the large interval of the included delay length.

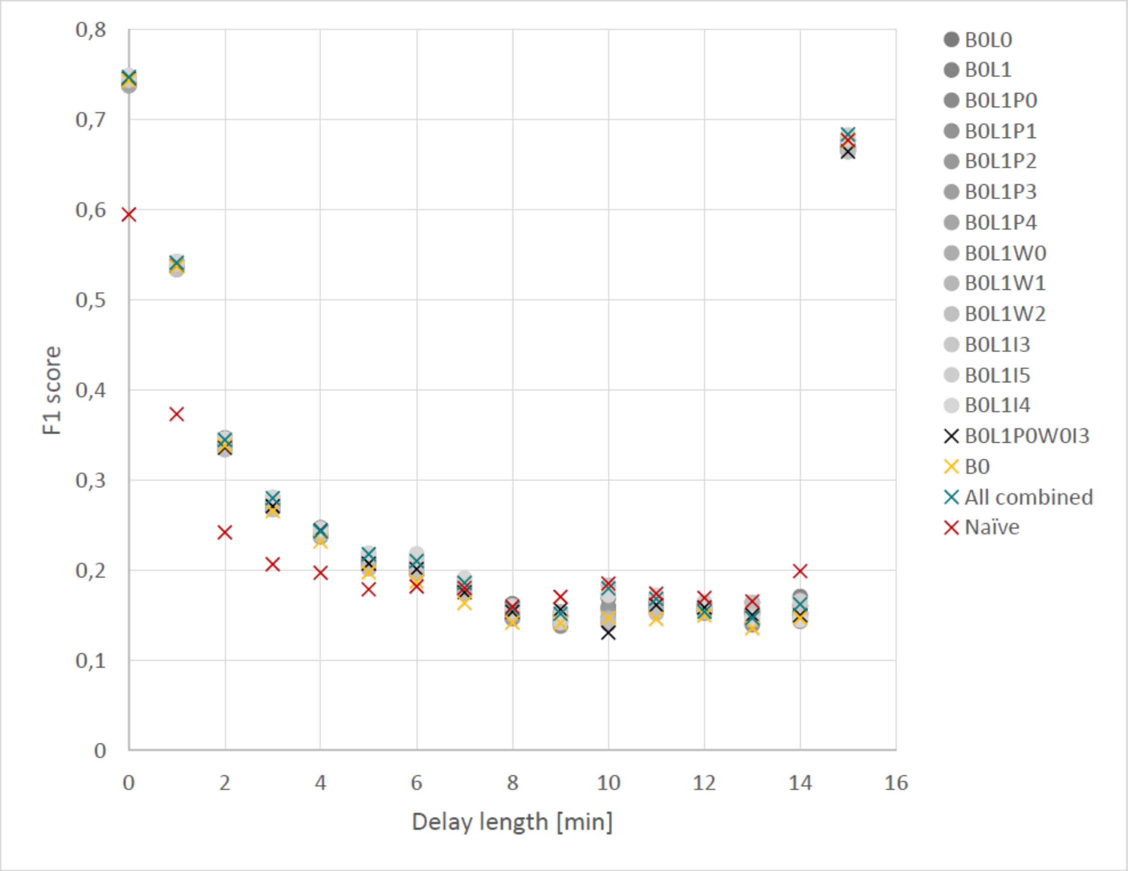


Figure 31 F1 scores of discretized delay prediction on the interval between 0 (and bellow) and 15 (and more) minutes of a delay.

It is important to keep in mind that the scores refer to precisely predicted corresponding delays. Depending on the application, a deviation up to a certain limit or in a certain direction might be acceptable. The percentage of delays being overestimated or underestimated is presented in Figure 32, although results are presented only for the Naïve forecast and the B0 feature set. That is because there is a clear visible difference, while the differences among the rest of the models are barely observable. The difference between the over- and underestimation of the 'Base' model and the 'all combined' model is in the order of 0.1 p.p.. The difference between the Naïve forecast and the 'Base' model shows larger changes (see Table 31). There is a significant decrease in delays overestimation (delays being predicted as larger than observed) but a slight increase in underestimation of larger delays (delays predicted as shorter than observed). The share of accurate predictions increases largely in the bin of not existing delays (including trains running ahead of the schedule). Then the increased percentage of accurate predictions continues up to delays of 10 minutes from where on, the accuracy decreases in comparison to the Naïve forecast.

Table 31 Change from the Naïve forecast to the B0 model in the share [%-points] of overestimated, accurate and underestimated delays within delay length bins.

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
overestimated	-23.9	-17.8	-11.6	-6.4	-4.3	-4.7	-7.7	-10.1	-10.4	-15.2	-13.0	-10.9	-10.2	-16.6	-14.5	0.0
accurate	23.9	10.5	6.2	7.1	6.8	4.7	2.9	1.7	0.1	0.4	0.1	-2.0	-1.6	-0.9	-5.6	-8.0
underestimated	0.0	7.3	5.4	-0.8	-2.5	0.1	4.8	8.4	10.4	14.8	12.8	12.9	11.8	17.5	20.2	8.0

Further improvement in accurate predictions (compared to the model of the B0 feature set) [%-points]

B0L1I3	0.4	0.1	0.7	1.6	2.0	2.1	2.4	2.8	1.6	0.6	2.7	3.0	1.9	0.9	2.0	1.2
all combined	0.3	0.5	0.7	1.3	1.0	1.7	2.1	3.0	2.0	1.3	3.7	3.0	0.5	0.8	1.8	1.4

Comparison of the same metrics among the various feature sets reveals that the highest percentage of accurate predictions across all the delay lengths is reached when train interactions features are included, and when the ‘Locations’ features are used. The accurate predictions then maintain higher percentage even with longer delays and the predictions fall behind the Naïve forecast only in the case of delays of 14 minutes and more. Accordingly, the overestimations and underestimations in the results of these feature sets are relatively lower comparing to other feature sets, which also means that the accurate predictions were captured relatively equally from both groups of over- and underestimated delays. Interestingly, another feature set performing well relatively to the other feature sets is the weather feature set version 2, which includes a feature reflecting ‘bad weather’ presence. Those results are generally comparable to the train interactions feature set in this case.

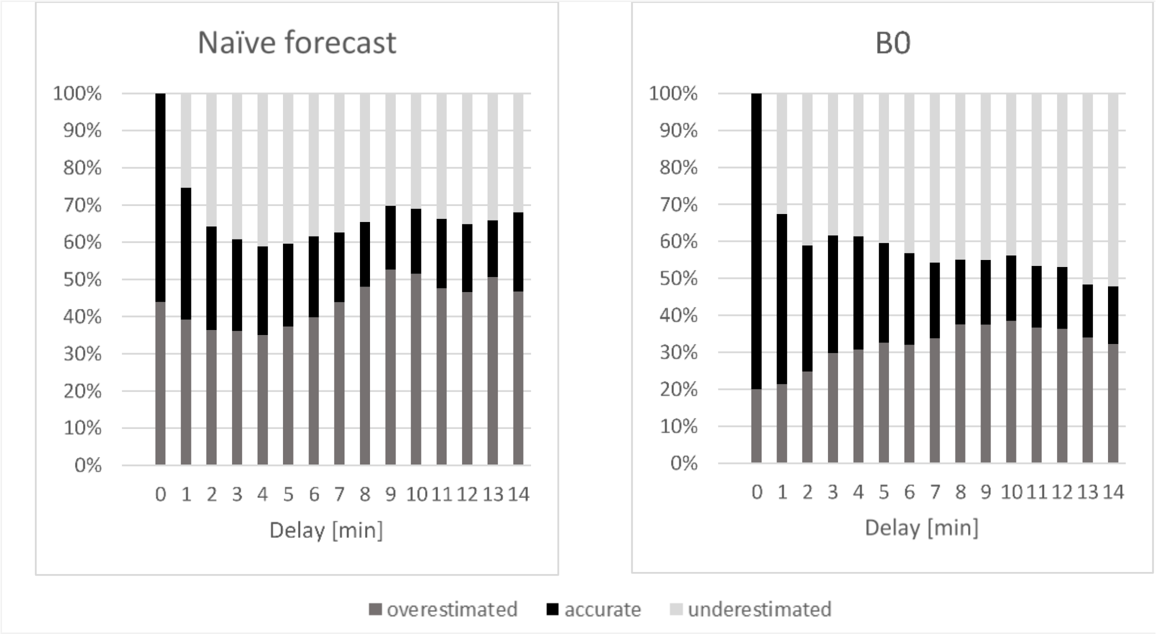


Figure 32 Accuracy of delay prediction: Comparison of Naïve forecast and a model built on the 'Base' feature set.

6.4. Classification tasks

Three classification tasks were defined in Section 3.5.1. To recapitulate and clarify the terms, the classification tasks are defined and will be referred to as follows:

Classification task	Classes	Notation	Description
Delay existence	*Positive*	1	a delay exceeding 2 minutes is predicted
	Negative	0	a not existing delay or a delay of maximum 2 minutes is predicted
Delay jump	*Positive*	1	a delay jump is predicted
	Negative	0	a not existing delay jump is predicted
Delay change	*Decreasing*	-	a delay is predicted to decrease by 2 or more minutes
	Constant	=	a delay is predicted not to change by 2 or more minutes in either direction
	Increasing	+	a delay is predicted to increase by 2 or more minutes

Representation of the classes is significantly disbalanced in our case as, for example, low delays predominating high delays and small delay changes predominating large delay changes. Reaching high scores in such highly frequented class is not any surprise and attention should be paid to scores of the less frequented class(es).

Delay existence

With respect to performance in predicting whether a delay will take place or not, the best performing feature sets are B0L1I3 and B0L1I4 if we look at the F1 score. Also the ‘All combined’ model reaches equal F1 score. Those are followed by B0L1P0W0I3, B0L1, and B0L0 and B0L1I5 in this order. An interesting observation is that weather feature sets lead to high scores in precision and lower scores in recall. Passenger features then show the opposite tendency. Precision in this context represents how many of the instances predicted to be delayed were also observed to be delayed. Low score in precision thus suggests that the delays tend to be overestimated. That is what the features reflecting passenger volumes seem to cause, because the precision scores are significantly worse compared to the simpler feature sets. Other than that, the precision scores are relatively equal among the other feature sets. As mentioned, recall noticeably drops when weather features are included (B0L1W0-2). That suggests delays are rather underestimated and thus some of the delays fall below the threshold of 2 minutes. Interestingly, low recall score is observed also with the B0L1P0W0I3 feature set, while it reaches the overall highest score in precision. High score in precision is achieved also by the ‘All combined’ model. See an overview of the scores in Table 32 and the complete results in Table 55 in Appendix B.3.

Comparing train interactions feature set variations with respect to the F1 score, the more complex versions (B0L1I0-2) perform worse than the simpler ones (B0L1I3-5). Also comparison of recall scores suggests the same conclusion. On the other hand, precision is higher when B0L1I0-1 feature sets are used. The feature sets B0L1I2 and B0L1I5 perform worse than the alternative feature sets, with an exemption of recall where they perform better than the other versions.

Table 32 Delay existence classification task results. (Positive class; See full results in Appendix B.3, Table 55)

Feature sets					F1	Precision	Recall	Accuracy
B	L	P	W	I				
Naïve forecast					0.586	0.462	0.798	0.898
0					0.655	0.548	0.814	0.909
0	0				0.659	0.551	0.819	0.911
0	1				0.660	0.553	0.819	0.911
0	1	0			0.657	0.552	0.812	0.910
0	1	1			0.654	0.545	0.816	0.910
0	1	2			0.653	0.545	0.815	0.910
0	1	3			0.654	0.546	0.815	0.910
0	1	4			0.653	0.545	0.815	0.910
0	1		0		0.656	0.559	0.795	0.909
0	1		1		0.655	0.558	0.793	0.908
0	1		2		0.655	0.558	0.793	0.908
0	1			3	0.665	0.559	0.820	0.912
0	1			4	0.665	0.559	0.820	0.912
0	1			5	0.659	0.550	0.822	0.911
0	1	0	0	3	0.661	0.564	0.799	0.910
All combined					0.665	0.561	0.816	0.912

Feature sets					F1	Precision	Recall	Accuracy
B	L	P	W	I				
0	1			0	0.655	0.552	0.806	0.904
0	1			1	0.655	0.551	0.808	0.904
0	1			2	0.647	0.537	0.813	0.903
0	1			3	0.656	0.549	0.813	0.904
0	1			4	0.656	0.549	0.813	0.904
0	1			5	0.648	0.537	0.815	0.903

Delay change

Expectedly, a more challenging task is to predict in which direction a delay will change, which is confirmed by the generally lower scores. Because we derived the delay change classification from the regression results, we did not predict directly the delay change. Instead, the classes represent whether the difference between the predicted and the present delay corresponds with the observed change or not. Referring to Table 23 in Section 4.6, decrease of delay occurs in 6.3% of the available observations and increase in 7%. The remaining 86.7% of observations show change of delay smaller than 2 minutes (in either direction). Therefore, there is a clear predominance of the 'constant class', which thus understandably scores well in this classification problem (with F1 scores around 0.93, precision 0.98 and recall approximately 0.88).

Low precision of a class implies decrease in recall of another class or classes. As can be seen in Table 33, precision of the 'decreasing' and 'increasing' classes are very low and recall of the 'constant' class is significantly lower than its precision. It is likely, that if an existing delay change (of any direction) is not predicted correctly, it is predicted as 'constant'. That was confirmed by an inspection of the data used for calculation of the metrics presented in Table 33. Especially low scores are reached in precision of the 'increasing' class. As observation of the data revealed, falsely predicting 'decreasing' class instead of the correct 'increasing' class is approximately twice as likely than vice versa. On the other hand, false predictions of observed 'constant' class are equally distributed between the two other classes. In general, predicted delay changes exceeding 2 minutes are with high probability actually smaller than 2 minutes, no matter the direction of the change. That means the predictions appear to overestimate delay changes. Predicted delay increase is correctly predicted only in approximately 13-16% cases and delay decrease in approximately 31-34% cases. Contrary to that, recall scores are relatively high, showing that decrease of delay is correctly predicted in approximately 67-70% and delay increase in about 49-58%.

Looking at differences among the feature sets, the highest F1 scores are reached by B0L1P0W0I3 and the 'All combined' model, followed by feature sets containing train interactions features and weather features. In accordance with the previous classification task, passenger counts features cause delay change overestimation leading to low precision and consequently F1 scores. On the other hand, weather features seem to have a significantly positive effect on delay change direction prediction, as the results of B0L1W0-2 are very comparable to the results of B0L1I3-4, which so far outperformed weather features. However, feature sets B0L1W0-2 lag behind in recall, especially in recall of the 'increasing' class, where they perform worse even than B0.

Among the feature sets B0L1I0-5, the F1 and precision scores are surprisingly higher in the more complex feature sets (B0L1I0-2) compared to their simpler alternatives. Conversely, the recall scores are higher when B0L1I3-5 are used and so is the overall accuracy. Predictions by models using the complex feature sets (B0L1I0-2) thus are more correct with respect to delay change identification, while the actual delay changes are correctly predicted with a slightly higher probability by the simpler feature sets (B0L1I3-5).

Table 33 Delay change classification task results. (Decreasing and Increasing classes; See full results in Appendix B.3,Table 56)

Feature sets					F1		Precision		Recall		Accuracy
B	L	P	W	I	-	+	-	+	-	+	
0					0.422	0.217	0.309	0.135	0.667	0.551	0.868
0	0				0.445	0.225	0.328	0.141	0.688	0.567	0.870
0	1				0.453	0.229	0.337	0.143	0.691	0.565	0.871
0	1	0			0.446	0.226	0.331	0.143	0.684	0.537	0.869
0	1	1			0.427	0.215	0.310	0.133	0.687	0.554	0.869
0	1	2			0.428	0.216	0.311	0.134	0.688	0.558	0.869
0	1	3			0.428	0.215	0.311	0.133	0.687	0.553	0.869
0	1	4			0.426	0.216	0.309	0.134	0.686	0.554	0.869
0	1		0		0.457	0.241	0.345	0.159	0.676	0.501	0.868
0	1		1		0.451	0.239	0.339	0.157	0.674	0.497	0.867
0	1		2		0.452	0.240	0.340	0.158	0.675	0.498	0.868
0	1			3	0.451	0.245	0.332	0.155	0.701	0.579	0.872
0	1			4	0.454	0.245	0.336	0.156	0.698	0.575	0.872
0	1			5	0.446	0.225	0.328	0.139	0.699	0.579	0.871
0	1			0	0.427	0.283	0.309	0.188	0.695	0.568	0.859
0	1			1	0.426	0.281	0.307	0.186	0.692	0.568	0.859
0	1			2	0.405	0.253	0.286	0.161	0.696	0.587	0.859
0	1			3	0.417	0.276	0.296	0.180	0.702	0.593	0.860
0	1			4	0.424	0.275	0.304	0.180	0.697	0.586	0.860
0	1			5	0.409	0.253	0.290	0.161	0.696	0.591	0.859
0	1	0	0	3	0.455	0.253	0.342	0.167	0.681	0.521	0.868
All combined					0.460	0.250	0.343	0.160	0.698	0.574	0.871

Delay jump

As was defined in Section 3.5, delay jump is a delay change equal to or exceeding 4 minutes, regardless direction of the change. According to that, delay jump can be observed in less than 3% of instances in the full dataset. That implies considerable disproportionality to the opposing class, which reaches equal scores across all the feature sets (F1 score 0.991, precision 0.999 and recall 0.984). Performance of the prediction with respect to this classification naturally cannot outperform combined prediction of the 'decreasing' and 'increasing' class above. Precision scores are therefore unsurprisingly low, predicting a delay jump correctly only in approximately 4-5% of the cases a delay jump is predicted. On the other hand, existing delay jumps are correctly predicted in about 37-47% of the cases. That means, delay jumps are highly overestimated, or in other words, delay changes are predicted to excess the threshold of a 4-minutes change. Inspection of the results revealed that approximately 10 times more delay jumps were predicted than were actually observed.

Accordingly to the results of the ‘delay change’ task, the highest F1 scores are reached by feature sets B0L1, B0L1I3-4 and B0L1W0. The latter feature set reaches the highest precision score, however, lags behind in recall and the overall accuracy. The most complex feature set B0L1P0W0I3 does not reach the highest performance especially due to poor recall. Contrary to that, the very highest recall score is attained by the ‘All combined’ model what contributes to the highest measured F1 score.

Table 34 Delay jump classification task results. (Positive class; See full results in Appendix B.3, Table 57).

Feature sets								
B	L	P	W	I	F1	Precision	Recall	Accuracy
0					0.072	0.040	0.390	0.982
0	0				0.083	0.046	0.442	0.983
0	1				0.090	0.050	0.449	0.983
0	1	0			0.084	0.047	0.405	0.983
0	1	1			0.072	0.039	0.412	0.983
0	1	2			0.071	0.039	0.411	0.983
0	1	3			0.073	0.040	0.418	0.983
0	1	4			0.071	0.039	0.409	0.983
0	1		0		0.090	0.051	0.379	0.982
0	1		1		0.089	0.050	0.390	0.983
0	1		2		0.086	0.049	0.372	0.982
0	1			3	0.090	0.050	0.470	0.983
0	1			4	0.090	0.050	0.457	0.983
0	1			5	0.087	0.048	0.473	0.983
0	1	0	0	3	0.088	0.050	0.393	0.982
All combined					0.091	0.050	0.476	0.983

Feature sets								
B	L	P	W	I	F1	Precision	Recall	Accuracy
0	1			0	0.065	0.036	0.356	0.980
0	1			1	0.063	0.035	0.361	0.980
0	1			2	0.051	0.027	0.352	0.980
0	1			3	0.057	0.031	0.393	0.980
0	1			4	0.052	0.028	0.374	0.980
0	1			5	0.055	0.030	0.373	0.980

Summary of classification tasks results

The three derived classification tasks we explored gave us additional information about the prediction quality. First, we tested whether solely existence of a delay in the future exceeding 2 minutes is correctly identified or not. Overall, about 79-82% of future delays were predicted and approximately 54-56% of predicted delays actually happened. Selecting the best performing feature sets based on the F1 score, those would be B0L1I3, B0L1I4 and ‘All combined’ model followed by B0L1P0W0I3 and B0L1. Passenger volumes features seemed to cause delay overestimation (at least around the threshold of 2-minutes delay) while weather features did vice versa.

Next, we assessed whether the change of delay between the predicted and the present delay corresponds with the observed delay change. The three classes to represent this problem were defined as either delay increase or decrease by 2 or more minutes and a constant delay. Delay decrease turned out to be correctly predicted with a higher probability, approximately in 67-70% cases, compared to 50-58% of correctly predicted cases of delay increase. Similarly, predicted delay decrease was correct in about 31-35% cases, while predicted increase only in about 13-16%. False predictions of the two classes in most cases fall to the ‘constant’ class with respect to both precision and recall. Nevertheless, delay change falling to ‘decreasing’ class instead of ‘increasing’ class was approximately twice as likely than vice versa. False predictions of observed ‘constant’ class were equally distributed between the two other classes. Generally, the predictions tend to overestimate delay changes. Finally, selection of the best performing feature sets according to the F1 scores yields choice of B0L1P0W0I3 and the ‘All combined’ model, followed by B0L1I3-4 and B0L1W0-2.

Eventually, the predictions were examined with respect to correct predictions of large delay changed of 4 minutes or more. Such delay changes are correctly predicted in approximately 37-47% of cases. Conversely, predicted delay jumps are actually observed in only about 4-5%. Large delay changes are therefore to a great extend overestimated. Considering the F1 scores of this task, the best performing feature sets are B0L1, B0L1I3-4 and B0L1W0, and the ‘All combined’ model.

In summary, the predicted delay change tends to be overestimated with respect to the defined thresholds of 2- and 4-minutes delay change. Over- and underestimation of the delay length affecting delay existence prediction can be seen in Figure 32. Delay change, however, does not necessarily correspond with that. Performance with respect to delay change therefore could not be directly retrieved from the regression results and thus the derived classification tasks provided us with additional information about the predictions performance. Finally, the best performing feature sets repeated throughout the three tasks, although with slight nuances. Nevertheless, the best performing feature sets in the three tasks were B0L1I3, B0L1I4, B0L1, B0L1P0W0I3 and the ‘All combined’ model. Finally, with respect to the delay change related tasks, the feature set B0L1W0, and possibly also B0L1W1 and B0L1W2, showed a significant positive effect on the precision scores.

6.5. Features importance

Metrics selection

In recapitulation, there are three feature importance metrics defined and provided with the XGBoost library (XGBoost Developers, 2016c). Those are *weight*, *cover* and *gain* as defined in Section 3.5.1. Shortly, *weight* represents the number of splits the feature was used for across all the trees, *cover* is the average number of times the splits on the feature were used, and *gain* is the average gain in the objective value on the feature’s splits. In other words, *gain* reflects the features’ contribution to the objective’s improvement. It therefore is an important metrics that relatively directly reflects the features’ importance in the model and is thus elaborated on in detail. In contrary to that, *weight* is rather an informative metrics than a metrics suitable for evaluation of the features’ importance. The total number of splits in a tree is limited by the maximum depth defined in the hyperparameters, and the total number of splits in a model is limited by the number of trees in the model. Consequently, more features in a model mean more features to ‘distribute’ the splits among. The features are selected in order to maximize gain at every split. A feature used for a split earlier in the tree perhaps brings higher gain but can appear on less splits than if used on the levels closer to the leaves and therefore attains lower scores on weight. On the other hand, a feature used close to the leaves likely brings less gain but can be used on more splits. A feature with a high score in weight therefore might have been used for splits close to the leaves, due to its lower contribution to the objective. Finally, *cover* says how many times the splits on the respective feature were actually used for the model’s development.

Presenting the scores individually for each model and train series is unreasonable and the average scores across the series are presented instead. However, the scores are dependent on the tree depth and the number of trees which both were variables that differ among the individual models. Therefore, to reduce the effects of these differences the scores were scaled to the highest score within each model (for every train series separately). The scores therefore represent average relative importance of the features. An interested reader can find all the scores in the Appendix 0. Scores of *gain* are presented also in Table 35 - Table 37 as they are commented on.

Disclaimer: Due to practical issues with results processing, feature importance metrics were not retrieved for feature sets B0L1I0-2 and B0L1P0W0I3.

Gain

Throughout the results, the present delay (‘Delay’) clearly reaches the highest scores in gain between 0.97 and 1.0, which means that the feature brings the most gain in a vast majority of the individual models, and especially when more features are present. In the B0 feature set, the leading feature is the direction (‘Direction’), and in the B0L0-1 feature set, the features with a considerably high gain were also the hour (‘Hour’), the upcoming departure locations feature (‘L0 / L1: V’) and the upcoming pass-through locations feature (‘L0 / L1: D’).

The remaining features have a rather supplementary role in the gain maximization. The next feature with relatively high scores in gain is the previous delay (‘Delay_1before’) scoring between 0.31 and 0.40. The following features such as the direction, a delay two locations back (‘Delay_2before’) and upcoming locations identification (‘L0 / L1: V’, ‘L0 / L1: K_V’, ‘L0 / L1: D’) then score between 0.1 and 0.2. However, when passenger counts or weather conditions are introduced, scores of the upcoming locations identification drop to or below 0.1. Train interactions do not cause such a drop. Nonetheless, the features of the B0 feature set show relatively stable importance across all the feature sets combinations. That is also due to the fact that it is always the present delay that reaches the highest gain and the remaining features of the set always score proportionally approximately equally.

Table 35 Gain scores of 'Base' and 'Locations' features.

	DOW	Location_nr	Direction	Delay	Hour	Minutes	Delay_1before	Delay_2before	L0 / L1: V	L0 / L1: K_V	L0 / L1: D
B0	0.05	0.11	0.17	0.97	0.07	0.08	0.31	0.10			
B0L0	0.05	0.08	0.14	0.99	0.08	0.06	0.40	0.12	0.14	0.12	0.13
B0L1	0.05	0.07	0.14	0.99	0.08	0.06	0.40	0.12	0.14	0.11	0.12
B0L1P0	0.05	0.07	0.13	0.99	0.08	0.06	0.33	0.11	0.13	0.10	0.12
B0L1P1	0.04	0.06	0.11	1.00	0.06	0.05	0.39	0.12	0.10	0.08	0.09
B0L1P2	0.04	0.06	0.11	1.00	0.06	0.05	0.39	0.12	0.10	0.08	0.09
B0L1P3	0.04	0.06	0.11	1.00	0.06	0.05	0.39	0.12	0.10	0.08	0.09
B0L1P4	0.04	0.06	0.11	1.00	0.06	0.05	0.39	0.12	0.10	0.08	0.10
B0L1W0	0.05	0.06	0.11	1.00	0.07	0.06	0.31	0.09	0.10	0.08	0.10
B0L1W1	0.05	0.06	0.11	1.00	0.07	0.05	0.38	0.11	0.10	0.09	0.10
B0L1W2	0.05	0.06	0.11	1.00	0.07	0.05	0.35	0.11	0.10	0.08	0.10
B0L1I3	0.04	0.07	0.13	0.99	0.07	0.05	0.36	0.11	0.13	0.10	0.11
B0L1I4	0.04	0.06	0.13	0.99	0.07	0.05	0.36	0.11	0.13	0.10	0.11
B0L1I5	0.04	0.07	0.13	0.99	0.07	0.05	0.36	0.11	0.13	0.10	0.11

Assessing the additional features, the upcoming locations identification features score relatively high (between 0.08 and 0.14) compared to the other features. Slightly higher scores are observed on the upcoming departure locations (‘L0 / L1: V’) followed by the upcoming pass-through locations features (‘L0 / L1: D’). All the features reflecting passenger counts (‘Pax_Total_1-4’, ‘Pax_Board_1-4’, ‘Pax_Alight_1-4’), the seats ratio (‘seats_ratio’) or the peak hour (‘peak_dep’) score comparably (0.04 or 0.05) although the relative gain of the peak hour decreases when the passenger counts features are included. That is understandable as the peak hour is partially implicit in the passenger counts features. The relative average gain of the features representing weather conditions resembles the scores of the passengers-related features, varying between 0.03 and 0.06. The highest score of 0.06 is obtained by the precipitation (‘Precp’) and temperature (‘Temp’), followed by the wind, view distance, rain and bad weather identification (‘Avgwind’, ‘Highwind’, ‘View’, ‘Rain’, ‘BadWeather’) scoring 0.05, and by the snow, storm, ice, and mist features (‘Snow’, ‘Storm’, ‘Ice’, ‘Mist’) scoring 0.04 or 0.03. Finally, the average relative gain of the train interactions features shows that the features representing interactions with empty rolling stock (‘LM’) are utterly powerless (scores of 0.00), and that identification of interactions with IC trains ('IC') scores relatively higher than interactions with SPR trains ('SPR') by scores of 0.04 or 0.05 compared to 0.03 and 0.04 respectively.

Table 36 Gain scores of 'Passengers' features.

	Seats_ratio	peak_dep	Pax_Total_1	Pax_Board_1	Pax_Alight_1
B0L1P0	0.05	0.05			
B0L1P1	0.05	0.04	0.05	0.05	0.05
B0L1P2	0.04	0.04	0.05	0.04	0.04
B0L1P3	0.05	0.04	0.05	0.04	0.05
B0L1P4	0.05	0.04	0.05	0.05	0.05

Table 37 Gain scores of 'Interactions' features.

	IC	SPR	LM
B0L1I3	0.04	0.03	0.00
B0L1I4	0.05	0.04	0.00
B0L1I5	0.03	0.03	0.00

Table 38 Gain scores of 'Weather' features.

	Avgwind	Highwind	Temp	Precp	View	Mist	Rain	Snow	Storm	Ice	BadWeather
B0L1W0	0.05	0.05	0.06	0.06							
B0L1W1	0.05	0.05	0.06	0.06	0.05	0.03	0.05	0.04	0.03	0.03	
B0L1W2	0.05	0.05	0.06	0.06	0.05						0.05

The average relative gain scores need to be interpreted in the context of the relevant feature set. However, to provide a comprehensive summary, average scores were calculated for each feature and those were ordered from the highest to the lowest (see Table 39). The order gives an idea of the features' contribution to the objective although regardless the specific variations of the features that were created. The variations were disregarded for this overview as there were no major differences observed.

Table 39 Average scores of the features' importance: Gain.

Feature	Average Score	Feature	Average Score	Feature	Average Score
Delay	0.99	Minutes	0.06	DOW	0.05
Delay_1before	0.37	Temperature	0.06	Pax_Board	0.05
Direction	0.12	View	0.05	Peak hour	0.04
L0 / L1: V	0.12	Average wind speed	0.05	IC	0.04
Delay_2before	0.11	Highest wind speed	0.05	Snow	0.04
L0 / L1: D	0.11	Rain	0.05	Storm	0.03
L0 / L1: K_V	0.09	Bad Weather	0.05	Mist	0.03
Hour	0.07	Pax_Total	0.05	SPR	0.03
Location_nr	0.07	Seats_ratio	0.05	Ice	0.03
Precipitation	0.06	Pax_Alight	0.05	LM	0.00

6.6. Visualization

All the tested feature sets were assessed by quantitative metrics, which provided us with means to compare the models' performance objectively and on a relatively detailed level. To understand the results from a more practical perspective and to develop a mental image of the predictions' behavior, several visualizations were created. First, the predicted and observed development of delays was visualized for which a set of four random trips of two train series were selected. The delays registered along those trips in time are depicted int Figure 33 (blue dotted line), along with the prediction of the final XGBoost model built on the B0L10I3 feature set prediction (solid green line) and the Naïve forecast (red dashed line). The B0L10I3 feature set was selected as a representation of the best performing feature set, which will be argued in the following section (6.7). As the prediction is available only from 20 minutes after the trips' beginning, the first 20 minutes of the trips are not depicted. It is possible to observe the delay underestimation originating from the nature of the Naïve forecast rigorously assuming the delays' decrease. Our prediction follows the line of the registered delay more closely although it is possible to see that there is a lag of approximately 20 minutes, which shows the ultimate importance of the present delay in the prediction (this is visible the best in the bottom right figure)

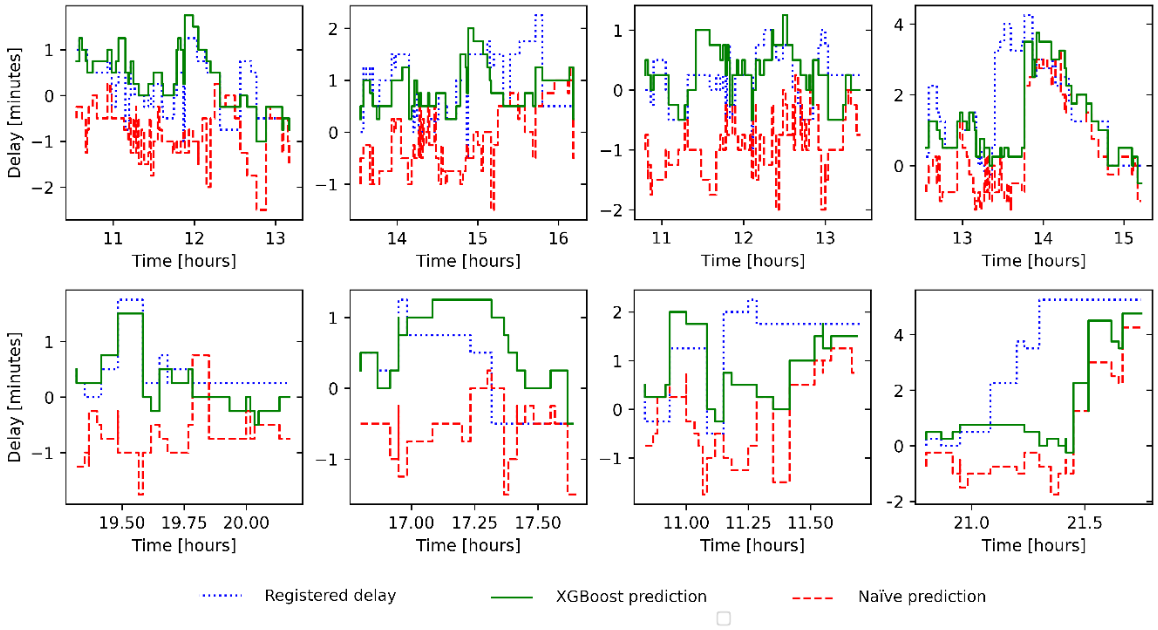


Figure 33 Delay development visualization.

The prediction and its errors distribution, and the delay length distribution can be seen in Figure 34 - Figure 39, showing the predictions by the Naïve forecast and the XGBoost models based on the B0 feature set and the finally selected B0L1I3 feature set. Figure 34, Figure 36 and Figure 38 show the full range of delays with a precision of 1 minute while Figure 35, Figure 37 and Figure 39 zoom in to delays on the interval between -5 and +15 minutes with a precision of 0.25 minute. The color range illustrates the number of observations belonging to the relevant pixels of the heatmap, where the x-axis is the registered delay and the y-axis is the predicted delay.

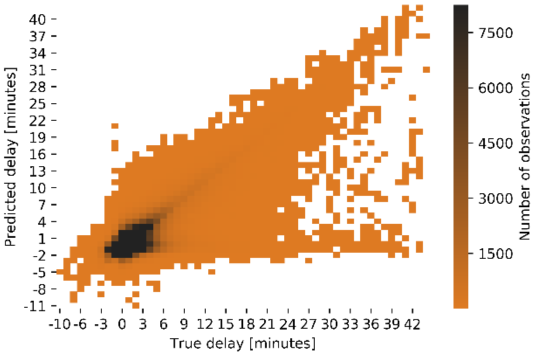


Figure 34 Heatmap of the true and predicted delay: Naive prediction; Full scale

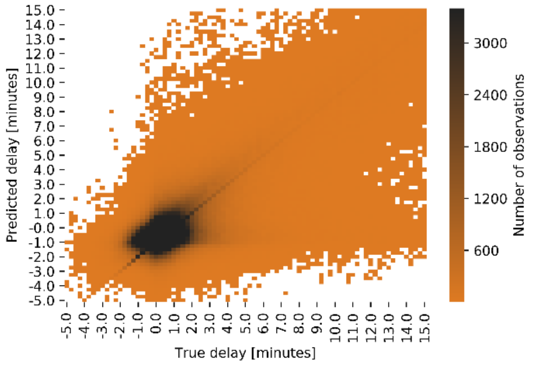


Figure 35 Heatmap of the true and predicted delay: Naive model; High resolution (0.25min); delays between -5 and +15 minutes

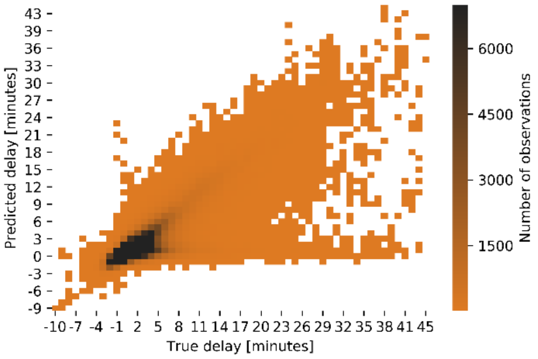


Figure 36 Heatmap of the true and predicted delay: "Base" model; Full scale

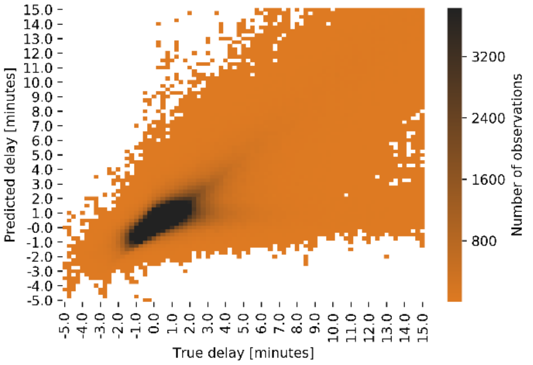


Figure 37 Heatmap of the true and predicted delay: "Base" model; High resolution (0.25min); delays between -5 and +15 minutes

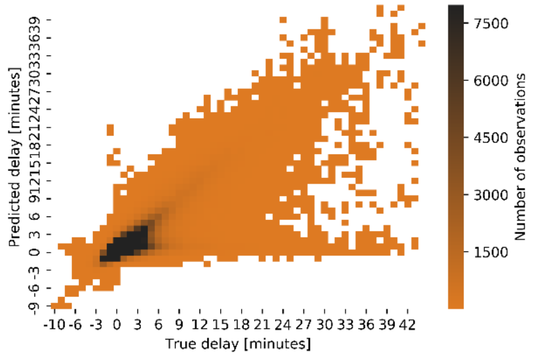


Figure 38 Heatmap of the true and predicted delay: "B0L1I3" model; Full scale

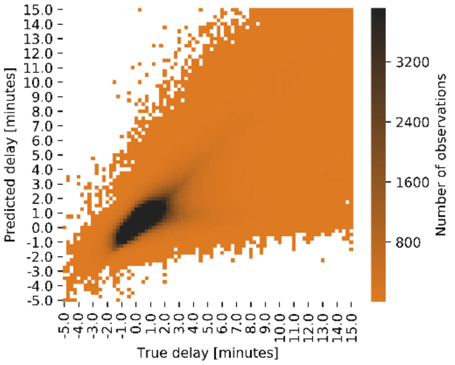


Figure 39 Heatmap of the true and predicted delay: "B0L1I3" model; High resolution (0.25min); delays between -5 and +15 minutes

Comparing the Naïve forecast and the basic XGBoost model, there is a clearly visible shift from the periphery towards the diagonal which is emphasized in the central area around the most frequent delays. Especially the area of negative predicted delays below the diagonal shows a significant shift towards reduction of underestimated delays. In the more detailed Figure 35, it is also possible to see the effect of the assumed 1-minute delay reduction, as there is a dark line just below the diagonal representing a set of observations that were underestimated by the 1 minute. All of the figures reveal an increased underestimation of delays just exceeding 2 or 3 minutes and a number of delays of any length that were predicted to be 0 minutes or nearly so. The assumption is that those are delays that were just registered and were not projected in the feature representing the present delay.

The shift in the numbers of observations in the confusion matrices behind the heatmaps is depicted in Figure 40 - Figure 43. The first two figures depict the shift from the Naïve forecast to the basic XGBoost model (B0 feature set), and the other two figures then show the shift from the basic XGBoost model to the final XGBoost model. Figure 40 depicts the difference in the number of observations falling to the individual cells of the matrix. Figure 41 then depicts the percentage this difference makes in the number of observations in the results of the basic XGBoost model. Especially the shift of the underestimated delays from below the diagonal is clearly visible. In Figure 41 the shift from the periphery along the diagonal is visible in the range up to approximately 7 minutes of delay. Furthermore, the decrease of falsely predicted negative delays is apparent and so is a shift in the predictions of the larger delays between 10 and 15 minutes, although there is no clear observable pattern.

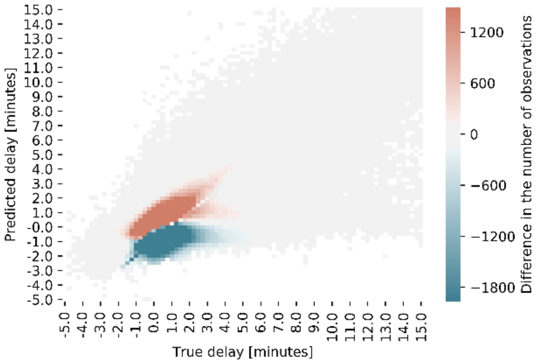


Figure 40 Heatmap of the shift of the counted true and predicted delay pairs: Comparing the Naive and the "Base" model (B0 featire set); High resolution (0.25 min); delays between -5 and +15 minutes

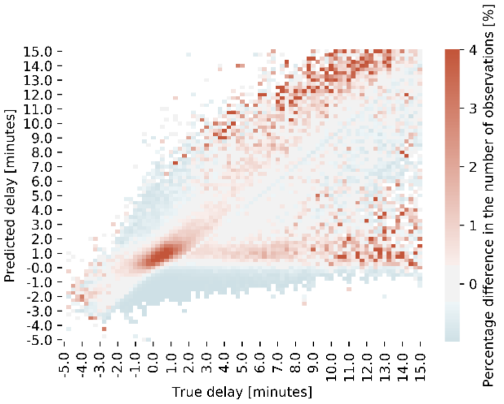


Figure 41 Heatmap of the shift of the counted true and predicted delay pairs in %: Comparing the Naïve and the "Base" model (B0 featire set); High resolution (0.25 min); delays between -5 and +15 minutes

The evaluation metrics did reveal an improvement by the final model compared to the basic model. Figure 42 displays the improvement in a more comprehensible way as it clearly shows a shift of a number of observations closer towards the diagonal. The number of those observations is relatively low in that region and therefore the change does not get depicted in Figure 43, which shows mainly shifts in the periphery due to the total lower number of observations there. Nevertheless, there is a visible improvement in the prediction by the final XGBoost model compared to the basic model.

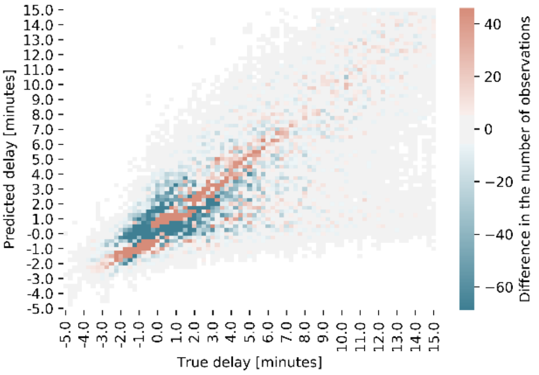


Figure 42 Heatmap of the shift of the counted true and predicted delay pairs: Comparing the "Base" and the "B0L1I3" model; High resolution (0.25 min); delays between -5 and +15 minutes

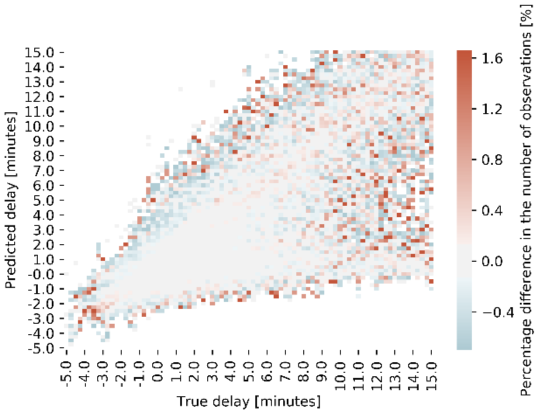


Figure 43 Heatmap of the shift of the counted true and predicted delay pairs in %: Comparing the the "Base" and the "B0L1I3" model; High resolution (0.25 min); delays between -5 and +15 minutes

6.7. Summary

The quality of the delay predictions done using the XGBoost model and a variety of feature sets were assessed from many perspectives to deeply analyze the predictions and develop understanding of its behavior. First, we shortly presented results of the XGBoost hyperparameters tuning. Although the hyperparameters are not necessarily directly relevant to the prediction results, it is an important component which contributes to the models' performance and it is an essential matter for the research's reproducibility. Next, we presented the core performance indicators derivable from regression results, RMSE and the related metrics. That was followed by the discretized prediction performance analysis, the derived classification tasks and finally feature importance assessment. Eventually, visual representation of the results was provided. All the performance indicators helped us to answer the final four research sub-questions.

Research sub-question(s) to be answered:

7. *What is the performance of the feature sets relatively to each other and to the reference model?*
8. *What is the importance of the features within the feature sets?*
9. *What features/feature sets have the highest/lowest potential in delay prediction?*
10. *What is the forecast potential of the highest performing feature set(s)?*

The results revealed that the highest improvement of performance comes from the basic feature set ('Base' or B0). That is mostly due to the feature representing the present delay which was found to bring the very highest gain in the objective. The basic feature set also contains other features that score high in the features' importance: the delays observed at the preceding registration points, the direction, hour and location identification, respectively. Next to those, the features representing the upcoming locations (feature sub-set L0 and L1) scored comparably high. That is in line with the observed prediction results as the feature sets B0L0 and especially B0L1 reached some of the highest scores in all the applied performance indicators.

Despite low scores in the features' importance scoring, train interaction features (creating feature sets B0L1I0-5 and B0L1P0W0I3) brought another significant improvement of the predictions, bringing the predictions to the highest scores with respect to all the performance indicators. Importantly, the improvement in the prediction performance was observed although the main comparison was done with the simpler versions of the train interactions features: based on train type instead of the more detailed feature sets based on train series (feature sets B0L1I3-5). Comparison of all the train interactions feature sets B0L1I0-5, done on the results of the models built for the subset of train series, revealed there might be space for further improvement of the predictions when the more complex feature sets are used. However, that is only due to the positive effect of those features on delay jump prediction where the feature sets B0L1I0-3 outperformed B0L1I3-5. As the delay jump is the most troublesome task, it is a viable and important improvement. On the other hand, the simpler versions (B0L1I3-5) performed better in all the other tasks, although the differences were slight. Regardless the features' complexity, the feature sets reflecting expected headway and expected violated headway scored both well, while the binary and quasi-binary identification of potential interactions scored lagged behind the alternative feature sets.

The features reflecting either passengers or weather conditions surprisingly did not bring as significant improvement, although there were certain slight improvements observable in some of the evaluation metrics. That was the case especially of the weather conditions, specifically the feature set B0L1W0. The B0L1W0-2 features had a significantly positive effect on delay change direction prediction although the feature sets lagged behind in recall, especially in recall of the 'increasing' class, where they performed worse even than the very basic feature set B0. B0L1W0 also appeared among the top performing feature sets with respect to delay jump prediction.

Out of the passengers feature sets, the simplest one (B0L1P0) performed the best, likely due to the seats ratio feature as was revealed in the features' importance analysis. Only negligible differences were observable among the other passengers feature sets variations (B0L1P1-4). Nevertheless, all the variations lead to deterioration of the results compared to the simper predecessor, the feature set B0L1. The features therefore caused confusion of the model instead of provision of useful information. Yet, in attempt to involve a representation of each feature category, also passengers features were chosen for the most complex feature set B0L1P0W0I3. Passengers features were represented by the simplest feature sub-set (P0) and similarly were the weather features (W0). The train interactions features could be represented by the simple sets only due to computational limitations. Out of those, it was the features exhibiting the expected headway to the first train belonging to the respective train type category that was selected (I3). Although this complex feature set performed relatively well, it did not outperform the feature set B0L1I3 which was thus selected as the final best performing feature set. It consisted of the following features:

- *'Base' sub-set*: Day of the week, Hour, Minute, Location, Direction, Current delay, Delay one and two registration points before
- *'Locations' sub-set* version 1: Total frequency of departures, short departures and pass-through points
- *'Interactions' sub-set* version 3: Expected headway to the first IC, SPR and LM train

Finally, the results largely varied among the individual train series. Every one of the feature sets performed the best for at least on train series. Some differences were negligible while some were substantially different. Implementation of such large variety of different models and features would significantly increase complexity of the application. Nevertheless, the best performing model (based on RMSE) of each train series was selected and the results were combined. Understandably, the overall results performed better in all the performance indicators compared to the other feature sets, including the finally selected feature set B0L1I3.

7. Conclusion and Discussion

In the beginning of our work, we discovered a research gap in 1) application of a gradient boosting decision trees-based model for delay prediction in passenger railways, 2) inclusion of passenger counts data in a passenger trains delay prediction and 3) combination of the three factors we incorporated (weather condition, passenger counts and train interactions) in the passenger railways delay prediction problem (see Section 1.1.2). Upon that, we defined an objective for our research: to define and evaluate sound feature sets reflecting effects of passenger counts, weather conditions and train interactions on passenger trains delays, which are to be used along with features derived from data available in the RAS competition (INFORMS, 2018e) to forecast the passenger trains' delays 20 minutes to the future. Effects of the individual features and their combinations are to be reflected upon (see Section 1.2).

Our work was carried out as a case study in cooperation with a passenger railway operator of the Netherlands, NS, and with support of the railway network manager in the Netherlands, ProRail. We used historical data from September 2017 to December 2017 to develop our models and test their performance. The data consisted of several datasets: planned timetable, realization data (AVL data), minimum required headways, passenger counts estimations and weather conditions measurement data. The data were used to derive features which were input to a delay prediction model based on the gradient boosting model XGBoost (T. Chen & Guestrin, 2016). The features were combined into a variety of feature sets of which the models' results were compared to the currently applied delay prediction method and among each other.

The starting point of our research was largely determined by research of Van den Bulk et al. (2018) who's study served as a benchmark for assessment of submissions to a competition that was announced by NS and ProRail within the RAS problem solving competition (INFORMS, 2019b) in 2018. Models proposed by Van den Bulk et al. and the two winning teams (Hellsten et al., 2018; Nabian et al., 2018) belonged to the same category: machine learning. The reccommended methods were either decision tree ensemble methods or neural networks. Highly comparable results did not favor any of the methods until findings of Van der Hurk (2019) lead to an inclination towards decision tree based methods. To closely build on work of Van den Bulk et al., we opted for the decision tree based gradient boosting method XGBoost they proposed, which was supported by the fact that the method has not been used for delay prediction in railway neither public transport yet.

We focused on effects of the various features we introduced on the predictions' quality. The results were compared among all the introduced feature sets and between the closely related feature sets for the relative performance assessment, and to the currently applied method, called Naïve forecast, for an absolute performance assessment.

In the following section (7.1), we guide the reader through the research sub-questions that gradually navigated our research towards discovery of an answer to the main research question, which is presented and answered in the end of section. Next, in Section 7.2, we discuss our research results and bring them into context with theory and literature. That is followed by a critical evaluation of our method in Section 7.3 and, finally, we list some recoommendations for further research in Section 7.4.

7.1. Conclusion: answers to the research questions

Our research sub-questions were formulated to guide our research from theory, through the application environment (the system and available data) to the model development and finally results evaluation. All the questions were answered in a broader context in summaries of the relevant chapters. Hereby, we present only the essential components of the answers instead.

1. *Why and how should the defined factors be included in the delay prediction model?*

The literature study revealed that all the three factors: passenger volumes, weather conditions and train interactions, play an important role in the railway system and can cause situations leading to train delay development. Thus, inclusion of the factors may indicate a higher probability of delay source presence or development. The form of the factors' inclusion in the model is determined by the model's nature. Therefore, the factors are defined in a form of features combined into feature sets. The passenger volumes were decided to be represented by a number of passengers scaled with respect to a certain representative number of passenger, which was inspired by research of Olsson & Haugland (2004). Weather condition data do not require any modifications and are directly retrieved from historical weather observations data. Finally, train interactions, which were subject of large amount of research in a wide variety of forms, are formulated upon inspiration gained from Carey & Kwieciński (1994), who emphasized the role of minimum headways between two consecutive trains, and Huisman & Boucherie (2001), who highlighted criticality of operations combining various train types. Combined with the type of train interactions data available for our research, the trains interactions are defined in a form of an expected headway (updated by the registered trains' delays) and its derivatives with respect to the train types.

2. *What are appropriate key performance indicators considering the model's characteristics?*

Multiple performance indicators were selected to provide us a variety of perspectives for the results assessment. The first group of performance indicators arises from the prediction task selection: regression. The first performance indicator is thus RMSE, because the algorithm in the model-building process evaluates the intermediate steps by this metrics. That is accompanied by MSE along with its confidence intervals. The other performance indicators were derived from those used in the RAS competition (INFORMS, 2019b): weighted RMSE and classification tasks performance metrics (accuracy, precision, recall and F1 score). For use of those, the regression results are transformed to a suitable form corresponding with the classification tasks defined in the RAS competition (delay jump and delay change) (INFORMS, 2019b) complemented by delay existence identification inspired by research of Fioole (2018b). Finally, assessment of the individual features' contribution in the models is done using feature importance indicators provided in the XGBoost library (XGBoost Developers, n.d.).

3. *How can/must the data representing the factors be modified to be used as features in the models expectedly maximizing their contribution to the model's performance?*

The train interactions data were transformed into features in the form of an expected headway and its derivatives. For every train, a set of potential interactions (due to sharing a part of the physical infrastructure) with other trains within the upcoming 20 minutes was defined together with identification of the involved trains. The planned headways and the minimum required headways were retrieved and updated by the delays of the involved trains. For every involved train series, only the first interaction was stored. The features were then defined per train series and per train type category. The passenger data could be directly used as a feature, however, scaling to reflect abnormalities rather than the total counts was opted for (as argued in Section 2.4.3). The selected scaling reflects various categories which may indicate the abnormalities with respect to the train number or series, location, day of the week or direction. The weather data did not require any transformation and could be directly used as features for the model. The only exception was the binary identified weather conditions, which were also transformed into a sum representing a presence of bad weather conditions.

4. *Is there an observable relationship between the included factors and train delays?*

The main pattern discovered by analysis of the train interactions features with respect to the future delay and delay change (delay change as the present delay is included as a feature on its own) showed a decreased probability of a delay reduction when the expected headway or expected violated headway reach certain periodical thresholds. Significance of the pattern varied among the pairs of interacting trains and train types. It was assumed that inclusion of the features could have a positive impact on overly optimistic predictions forecasting delay decrease, although with a different notability for the various train series. Concerning the passengers-related features, there is an observable increase in probability of existence of larger delays when the realized capacity is smaller than the planned capacity was. The relation between the delay change and the number of passengers or the scaled numbers of passengers yields a tendency of a decreased delay change in either direction with an increasing (or in a smaller extent also a decreasing) number of passengers. Although the differences among the features' versions appear to be negligible, we proceeded with all of them to the modelling part to confirm or deny this hypothesis. Finally, the weather features were paired with the corresponding delay changes and analyzed. It showed an increase in delay growth when high wind speed was measured, decreased delay change with an higher levels of precipitation and decreased delay change with larger deviation of the temperature from the most frequently observed temperature. The binary identified weather conditions were present rather sparsely, except for the rain and mist. Presence of mist did not seem to have a significant impact on the delay change, but presence of rain suggested increase in delay and so did presence of snow and storm.

5. *What are sound feature sets?*

The features were grouped into feature sub-sets by the category they were affiliated with (e.g. passenger volumes) and their version. Those feature sub-sets are to be combined into a number of feature sets, where the number of feature sets was highly dependent on feasibility of the study due to time and computational limitations. All of the feature sets contain the '*Base*' sub-set which matches the basic feature set of Van den Bulk et al. (2018). That is accompanied by the '*Locations*' features indirectly representing likeliness of delays induced by busyness of the upcoming network section. Those feature being derived from the timetable as the 'Base' was, they become a permanent part of the basic set in the further expansion. Next, all the versions of the feature sub-sets of the remaining categories are individually added. The best performing version of each category is to be selected and those combined into a complete feature set consisting of sub-set from all the categories.

6. *What is a sound model structure including parameter tuning?*

An algorithm was developed to build a model for each train series separately, which was done following research of Van den Bulk et al. (2018). As the hyperparameters optimization is a time-consuming step, it is performed only once for each feature set: optimized on the first in order train series. A set of the best performing hyperparameters combinations is then used as a set of candidates for the remaining train series. The hyperparameters tuning is done using a training and validation set. The final model is trained on the train set and a prediction is made on the test set.

7. *What is the performance of the feature sets relatively to each other and to the reference model?*

The highest improvement of performance comes from the basic feature set (B0), which, compared to the Naïve forecast, decreases the RMSE from 1.99 to 1.52, and decreased width of the MSE 95% confidence interval by 25%. The features representing the upcoming locations in the feature sub-sets L0 and L1 further decreased the RMSE by 0.01 and 0.02 respectively (RMSE 1.50 and 1.49). The passengers features added to the B0L1 feature set caused increase of the RMSE by 0.01 or 0.02 and so did the weather features. The train interactions features on the other hand caused an improvement by decreasing the RMSE by 0.01 (RMSE 1.48), except for the last version (I5) which did not change the RMSE but shifted the MSE 95% confidence interval by -0.01. Comparing differences between all the train interactions (tested on a subset of train series), the lowest RMSE is reached with version I3 and I4, followed by I5 with the RMSE higher by 0.01, I0 and I1 with an additional increase of the RMSE by 0.01 and finally I2 with another increase by 0.01 on the RMSE. Finally, the feature set consisting of feature sub-sets of all the categories is burdened by a RMSE of 1.50. Results of the underlying individual models built for the separate train series largely vary among the feature sets and train series and every feature set performs the best for at least one train series. Nevertheless, the most frequently best-performing feature sets contain train interactions features and/or weather features. Performance of the feature sets with respect to the remaining performance indicators follows relatively the same pattern. However, weather features bring an improvement in delay change and delay jump prediction, especially in the prediction precision.

8. *What is the importance of the features within the feature sets?*

The features were assessed by a feature importance metric called gain, which was then scaled to the highest observed value in the respective model. The present delay reached the highest scores in gain in a vast majority of the individual models. Next highly scoring features were the direction, hour, the upcoming departure locations and the upcoming pass-through locations feature and the previous delays. However, considering the predictions' performance results, the feature sets including train interactions features achieved the highest scores and thus the features must have a significant and positive impact on the prediction. Comparison of performance among the respective feature set variations, the binary and the derived quasi-binary versions (sum of the underlying binary features) are significantly outperformed by those representing expected headways and expected violated headways. Preference between those two is not so straightforward and the differences are rather negligible, although with a slight preference towards the first version. Comparison of the features being defined per train category or train series also shows rather small nuances, however, the latter case features have a positive impact on delay jump prediction, which is a viable argument in their favor. Finally, the weather features showed a positive impact on precision of prediction of delay change direction and delay jump. That means, the features inclusion resulted in smaller expected delay changes, which on the other hand lead to overestimation of the delay changes and decreased recall. Nevertheless, judgement of its benefit depends on subjective preferences. Supposedly the impacts are viable, it is likely the precipitation, temperature, view distance, wind speed and rain and bad weather identification that has the largest impact.

9. *What features/feature sets have the highest/lowest potential in delay prediction?*

In reference to the two preceding questions, the best performing feature set in our research consists of the basic sub-set, the locations sub-set version 1 and train interactions sub-set version 3 (B0L1I3). That implies the feature set is composed of the following individual features:

- 'Base' sub-set: Day of the week, Hour, Minute, Location, Direction, Current delay, Delay one and two registration points before
- 'Locations' sub-set version 1: Total frequency of departures, short departures and pass-through points
- 'Interactions' sub-set version 3: Expected headway to the first IC, SPR and LM train

In addition to that, some features from the other categories have a potential in delay prediction and those are certain weather conditions features, especially precipitation and temperature, possibly also view distance, wind speed and rain or bad weather identification.

10. *What is the forecast potential of the highest performing feature set(s)?*

Compared to the Naïve forecast, the feature set B0L1I3 (see the preceding questions) brings an improvement in the RMSE of 0.51, thus the RMSE changes from 1.99 to 1.48, or likewise the MSE changes from 3.94 to 2.20. The MSE 95% confidence interval shifts from [3.90, 3.98] to [2.17, 2.23]. The feature set reaches the lowest RMSE by 29% of train series, which is the highest share among the feature sets. With respect to delay existence prediction, the F1 score of the Naïve forecast is 0.586, while the selected feature set reaches a score of 0.665. Also the precision score shows an improvement (from 0.462 to 0.559) and so does the recall (from 0.798 to 0.820) and accuracy (from 0.898 to 0.912). Delay change and delay jump prediction was not evaluated for the Naïve forecast due to its nature of rigorously 'predicting' delay decrease by 1 minute. Nevertheless, the F1 scores of the decreasing and increasing classes of delay change prediction by the finale feature set are 0.451 and 0.245, respectively. Recall scores of the classes are 0.332 and 0.155, respectively, and precision scores 0.701 and 0.579 in the corresponding order. The F1 score of delay jump prediction then is 0.090 with precision of 0.050 and recall 0.470.

The research sub-question had led us throughout the entire process towards the climax of our work, the conclusion formulated as an answer to the main research question, which was formulated as follows:

Research question

What is the performance of passenger trains' delays forecast 20 minutes to the future if a machine learning model, built on previous research by NS and extended by passenger and weather data and offline identified and online updated train interactions, is used, and how do these factors contribute?

In a close reference to the questions answered above, the best performing model was not the one built on the most complex feature set consisting of features from all the categories. The most complex feature set (B0L1P0W0I3 consisting of feature sub-sets base, locations v.1, passengers v.0, weather v.0 and interactions v.3, see the feature sub-sets description in Section 5.1) was burdened by a RMSE 1.50 (correspondingly by a MSE 2.25 with a 95% confidence interval [2.21, 2.28]) and RWMSE 1.36, while the finally selected feature set (B0L1I3 consisting of feature sub-sets base, locations v.1and interactions v.3) reached RMSE 1.48 (correspondingly a MSE 2.20 with a 95% confidence interval [2.17, 2.23]) and RWMSE 1.35. Although the B0L1P0W0I3 feature set outperformed most of the other feature sets in delay change prediction, it lagged behind with respect to all the other performance indicators. According to the results of models consisting passengers features and the feature importance analysis, it is the passengers features that possibly worsen the prediction performance. On the other hand, the train interactions features and some of the weather features do contribute to the prediction quality, according to the applied performance indicators. The train interactions features have a strong improving effect on the performance with respect to all the performance indicators, while weather features affect especially the resulting delay change prediction.

7.2. Discussion

Our research was initiated upon practice-driven interest of the railways operator in the Netherlands, NS, and fitted to an identified research gap. As defined in Section 1.1.2 and recapitulated at the beginning of this chapter, the research gap was found in three aspects, according to which we will discuss the results of our research. In addition, we will compare our results to results of Van den Bulk et al. (2018), whose research substantially influenced ours.

1. *Application of a gradient boosting decision trees-based model for delay prediction in passenger railways.*

As we discussed in the literature review, most of the recent studies concerning delay prediction in passenger railways applied neural networks, Bayesian networks, shallow and deep extreme machine learning, random forest or ensemble methods combining multiple approaches. For a valid and truly meaningful comparison of performance of different approaches, the methods should be applied under the same conditions: applied to the very same problem, using the same data and features definition (if possible). The RAS competition (INFORMS, 2018e) was a creative way to approximate such conditions and explore a broad range of methods, although employment of the provided data did vary and so did depth of the methods application. Nevertheless, as mentioned in Section 3.3, the winning approaches were neural network and decision tree-based methods. Due to close comparability of the prediction performance, there was no clear preference towards either of the two machine learning branches. Selection of XGBoost was eventually supported by findings of Van der Hurk (2019). Due to the time limitations of our work, we could not test various approaches, especially as our focus was paid to exploration of a wide range of features instead. Nevertheless, our models did bring a significant improvement in the prediction compared to the baseline. For example, the RMSE dropped from 1.99 to 1.48, which is a significant improvement meaning that an average error of the prediction is approximately 88 second compared to nearly 120 seconds. Considering that the trains were registered with a delay between -1 and 1 minute in 58.3% of the registrations, and between -2 and 2 minutes in 80.6% of the registrations, an average error of 120 seconds is a significant inaccuracy and the decrease of the error of approximately 32 seconds is therefore a considerable improvement. Along with conclusions arising from the other performance indicators, this proves that the applied method does have a potential of improving delay prediction.

2. *Inclusion of passenger counts data in a passenger trains delay prediction.*

To our surprise, we found a large gap in the literature in application of passenger counts for delay prediction in railway. Contrary to that, there is a considerable amount of literature applying passenger counts for delay prediction in bus operations. However, due to the differences between the two systems (such as differences in fare collection and thus retrieving passenger counts in real time; and often significantly different boarding and alighting characteristics), the findings cannot be easily extrapolated to railway operations. Upon findings of Olsson & Haugland (2004), who found correlation between departure punctuality in passenger railway and the number of onboard passengers related to the highest number of passengers on the relevant train, we decided to relate the passenger counts used in our work to a certain value believed to represent a regular number of passengers under specified temporal and local conditions. Differences between those variations were slight and so was the observed relation to delay and delay changes, nevertheless, we decided to proceed with them to the modelling and performance assessment phase. Results confirmed that the features (in the form as defined in our work) do not have any prediction power and rather introduced noise in the predictions. There was no performance indicator, where the features would cause any improvement and only nuances in the results' deterioration were observable. Furthermore, there were no observable differences in the results comparing the features variations. Neither the ratio of planned and realized seats brought any improvement. We therefore conclude that inclusion of passenger counts into train delay prediction model in the context of our work is unprofitable and rather disruptive. That is in contrary to the large amount of literature proving the role of passenger volumes on train delay development. However, we used passenger counts measured at a past location for a future delay prediction. Thus, it is likely that the possibly unusual passenger volumes caused a delay change implicit in the train's present delay. Its effect in the future is then negligible. That is despite the fact that an unusually high volume of onboard passengers has to exit the train at some point and thus possibly cause an extended alighting time.

3. *Combination of the three factors we incorporated (weather condition, passenger counts and train interactions) in the passenger railways delay prediction problem*

Referring to the previous paragraph, use of passenger volumes for train delay prediction was generally lacking in the literature. On the other hand, train interactions were a subject of an excessive amount of relevant literature. Finally, weather condition appeared in a few approaches, yet it was not a widely used factor in delay prediction. As defined in Section 2.2, passenger volumes and weather are related to potential causes of primary delays, while train interactions refer rather to secondary delays. In the reviewed literature, researchers frequently build ensemble models consisting of a part dedicated to delay propagation due to train interactions. We wanted to test if train interactions can be defined the same way as potential primary delay sources. In that way, the resulting features behave as indicators of situations with an increased risk of delay induction instead of predicting delay propagation in a deterministic matter. Naturally, this leads to increased uncertainty and noise in the results due to relaxed interpretation of the features. Even the very simplified features representing expected headways to the time-wise nearest train causing an interaction (sharing a piece of the physical infrastructure) resulted in a significant improvement of the delay prediction. Referring to the RMSE, we observed, for example, an improvement by 0.01 (from 1.49 to 1.48) compared to the preceding feature set. The MSE then correspondingly dropped from 2.23 with a 95% confidence interval [2.20, 2.27] to 2.20 with a 95% confidence interval [2.17, 2.23]. Definition of the features was highly simplified and burdened by a number of assumptions, yet had a positive impact. Together with the low computational efforts necessary to retrieve such features, there is a potential for application of similarly defined train interactions features. Finally, as discussed in the research sub-questions, weather features did have an effect on the predictions, especially by reducing expected delay change. That resulted in an increased precision, however, also significant deterioration of the recall scores associated with delay change and delay jump prediction. In summary, combining primary and secondary delay indicators in a single model was found to be successful and we expect there is space for further development of such an approach.

4. *Reference to research results of Van den Bulk et al. (2018)*

We referred to the work of Van den Bulk et al. (2018) throughout our work, as they set the starting point and the main direction of our work. As our models were developed independently, it is highly likely that there were differences in data processing, features definition and model development. Although our base feature set was composed according to the basic feature set of Van den Bulk et al. (2018), the results slightly differ. The RMSE obtained by Van den Bulk et al. (2018) and using the basic feature set is 1.37 while results of our corresponding model show a RMSE 1.52. Van den Bulk et al. (2018) developed two additional feature sets reflecting train composition changes and driver changes, and highly simplified train interactions. The latter model was hindered by too many features and did not perform well. The earlier model decreased the RMSE to 1.34, thus by 0.03 or 2.2 %. Or best performing model resulted in an improvement in the RMSE by 0.03 as well, thus 2.0 %. Van den Bulk et al. (2018) did not perform a delay jump prediction, only delay change. They present the results aggregated for the two categories of delay increase and decrease. Nevertheless, they observed worsening of the precision, constant recall and constant F1 score comparing the basic model and the second model of theirs. Contrary to that, we observed rather constant precision scores and improved recall scores comparing the base feature set and the best performing models.

7.3. Limitations

In this section, we identify limitations of several aspects of our approach and shortly comment on each.

XGBoost hyperparameters tuning: We developed a model for every train series separately (in total 79 train series). Optimization of the hyperparameter for every individual model would be excessively time consuming. Therefore, we optimized the hyperparameters once for every feature set category. We stored a set of the best performing hyperparameters combinations, which were used as candidates for all the forthcoming models. Nevertheless, there was a risk, that the best suiting hyperparameters for certain models were not available among the candidates.

Weather features: We used historical weather measurement data. For the prediction, data from the present hour were used. For application in real time, the current weather condition would have to be processed into the features. Instead, weather forecast would be possibly applicable, as was used in research by Oneto et al. (2016, 2017, 2018).

Passenger features: The passengers data are an estimation based on smart card data, which is computed retrospectively. Gaining the same data in real time is therefore hardly possible. Furthermore, passenger data are generally burdened by a strict non-disclosure policy, which puts additional constraints on manipulation with the data.

Train interactions features: Process of the features derivation was highly complex, and many assumptions and simplifications had to be done for practical reasons, because data originating in two different companies (NS and ProRail), and thus not entirely consistent, had to be merged. In result, some interactions may have been overlooked or misinterpreted. However, the major limitation of the train interactions feature sets lies in the fact, that they account for the present state of the network only, with no further consideration of any interdependencies among the other trains. Furthermore, the features and nature of the model do not guarantee a correct prediction of an apparent delay propagation situation (e.g. an IC train caught behind a SPR train).

'All combined' model: The models were developed per train series and their scores highly varied among the feature sets. Every feature set was preferable for at least one train series. To maximize the prediction performance, we combined the best results (according to the RMSE) of every train series which indeed lead to an improved prediction on the network level. Although the difference was limited, it points at the fact that there are differences among the train series and a uniform approach might be suppressing the full potential of the predictions. Using various feature sets however introduces a risk of considerable complexity in potential application.

Data splitting: The data was split for training, testing and assessment set by train trips to avoid overfitting. Yet, there is a considerable concern that presence of multiple train registrations (together with the attached additonal features) often shortly after each other affects the the training process (by presence of multiple highly similar instances) as well as the results (prediction being done for multiple very similar instances).

Performance assessment: The main limitation is associated with the derived classification tasks. Those were initially designed as simper probems compared to regression prediction. We used them as an additional tool for analysis of the predictions' behavior. It however does not reflect how the feature sets would perform when used for real classification tasks.

7.4. Recommendations for further research

In reference to the limitations and the results of our research, there is a list of recommendations we may give for further research. First, we address characteristics of the prediction. Instead of making the prediction from a registration point, it could be done from a time point which would require to consider the meanwhile traveled part of the link between the last and the next registration point. Next, the prediction time window of 20 minutes can be changed and development in the prediction performance observed as the prediction time window changes. Another suggestion is to focus on predicting delay change instead of the actual future delay.

Regarding the model characteristics, a different evaluation metrics than RMSE may be used in the XGBoost model. Next to a set of pre-defined evaluation metrics, the model allows to use user-defined evaluation metrics. Above that, the model allows deeper configuration setting and we recommend exploring the full possibilities the XGBoost library offers. Special attention then can be paid to the hyperparameters optimization for which we recommend using a different method than genetic algorithm.

Concerning features, we recommend exploring possible definition of train interactions features. Those we defined were highly simplified, yet they brought improvement to the prediction. There is virtually boundless amount of possible definition of train interactions by the means of features, and it is probable some of them have a high potential to improve the prediction even further. Next, apparent delay propagation situations (e.g. an IC train caught behind a SPR train), that are overlooked by the model but are possible to be defined otherwise, can be accounted for by supplementary adjustment. The amount of such definable occurences can be observed and the effect on the final prediction evaluated.

8. References

Albert, S., Kraus, P., Müller, J. P., & Schöbel, A. (2017). Passenger-Induced Delay Propagation: Agent-Based Simulation of Passengers in Rail Networks. *Simulation Science*, 3–23. https://doi.org/10.1525/california/9780520292765.003.0006

Anderson, M., Antenucci, D., & Bittorf, V. (2013). Brainwash: A Data System for Feature Engineering. *Eecs.Umich.Edu*. Retrieved from http://web.eecs.umich.edu/~mrander/pubs/mythical_man.pdf

Au, T. C. (2018). Random forests, decision trees, and categorical predictors: The "absent levels" problem. *Journal of Machine Learning Research*, *19*, 1–30.

Bai, J., & Ng, S. (2005). Tests for skewness, kurtosis, and normality for time series data. *Journal of Business and Economic Statistics*, *23*(1), 49–60. https://doi.org/10.1198/073500104000000271

Bates, J., Polak, J., Jones, P., & Cook, A. (2001). The valuation of reliability for personal travel. *Transportation Research Part E: Logistics and Transportation Review*, *37*(2–3), 191–229. https://doi.org/10.1016/S1366-5545(00)00011-9

Bell, M. G. H., & Iida, Y. (1997). Network Reliability. In *Transportation network analysis* (J. Wiley). New York: Chichester.

Berger, A., Gebhardt, A., Müller-Hannemann, M., & Ostrowski, M. (2011). Stochastic Delay Prediction in Large Train Networks. In *11th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems* (pp. 100–111). https://doi.org/10.4230/OASIcs.ATMOS.2011.100

Bhandari, D., Murthy, C. A., & Pal, S. K. (2012). Variance as a stopping criterion for genetic algorithms with elitist model. *Fundamenta Informaticae*, *120*(2), 145–164. https://doi.org/10.3233/FI-2012-754

Biau, D. J., Jolles, B. M., & Porcher, R. (2010). P value and the theory of hypothesis testing: An explanation for new researchers. *Clinical Orthopaedics and Related Research*, *468*(3), 885–892. https://doi.org/10.1007/s11999-009-1164-4

Börjesson, M., & Eliasson, J. (2011). On the use of " average delay" as a measure of train reliability. *Transportation Research Part A: Policy and Practice*, *45*(3), 171–184. https://doi.org/10.1016/j.tra.2010.12.002

Breiman, L., Friedman, J. H., Olshen, R. A., & Stone, C. J. (1984). *Classification and Regression Trees*. Belmont: Wadsworth International Group.

Brouwer, J. (2012). *Delay prediction model: Using historical data to predict delays in railways*. Utrecht.

Cambridge University Press. (2019a). Meaning of delay in English. Retrieved October 10, 2019, from https://dictionary.cambridge.org/dictionary/english/delay

Cambridge University Press. (2019b). Meaning of punctuality in English. Retrieved October 10, 2019, from https://dictionary.cambridge.org/dictionary/english/punctuality

Carey, M. (1999). Ex ante heuristic measures of schedule reliability. *Transportation Research Part B: Methodological*, *33*(7), 473–494. https://doi.org/10.1016/S0191-2615(99)00002-8

Carey, M., & Kwieciński, A. (1994). Stochastic approximation to the effects of headways on knock-on delays of trains. *Transportation Research Part B*, *28*(4), 251–267. https://doi.org/10.1016/0191-2615(94)90001-9

Cerreto, F., Nielsen, B. F., Nielsen, O. A., & Harrod, S. S. (2018). Application of Data Clustering to Railway Delay Pattern Recognition. *Journal of Advanced Transportation*, 1–18. https://doi.org/10.1155/2018/6164534

Chai, T., & Draxler, R. R. (2014). Root mean square error (RMSE) or mean absolute error (MAE)? -Arguments against avoiding RMSE in the literature. *Geoscientific Model Development*, *7*(3), 1247–1250. https://doi.org/10.5194/gmd-7-1247-2014

Chen, M., Liu, X., Xia, J., & Chien, S. I. (2004). A dynamic bus-arrival time prediction model based on APC data. *Computer-Aided Civil and Infrastructure Engineering*, *19*(5), 364–376. https://doi.org/10.1111/j.1467-8667.2004.00363.x

Chen, M., Yaw, J., Chien, S. I., & Liu, X. (2007). Using automatic passenger counter data in bus arrival time prediction. *Journal of Advanced Transportation*, *41*(3), 267–283. https://doi.org/10.1002/atr.5670410304

Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 785–794). San Francisco. https://doi.org/10.1145/2939672.2939785

Chopde, N. R., & Nichat, M. K. (2013). Landmark based shortest path detection by using Dijkestra Algorithm and Landmark based shortest path detection by using Dijkestra Algorithm and Haversine Formula. *International Journal of Innovative Research in Computer and Communication Engineering*, *1*(2).

Corman, F., & Kecman, P. (2018). Stochastic prediction of train delays in real-time using Bayesian networks. *Transportation Research Part C: Emerging Technologies*, *95*, 599–615. https://doi.org/10.1016/j.trc.2018.08.003

Cornet, S., Buisson, C., Ramond, F., Bouvarel, P., Rodriguez, J., Cornet, S., … Rodriguez, J. (2019). Methods for quantitative assessment of passenger flow influence on train dwell time in dense traffic areas. *Transportation Research Part C*, *106*(March), 345–359. https://doi.org/10.1016/j.trc.2019.05.008

Cox, E. (2005). *Fuzzy modeling and Genetic algorithms for data mining and exploration*. (J. Gray, Ed.), *The British Journal of Psychiatry*. Morgan Kaufmann Publishers is an imprint of Elsevier.

Cox, T., Houdmont, J., & Griffiths, A. (2006). Rail passenger crowding, stress, health and safety in Britain. *Transportation*

Research Part A: Policy and Practice, *40*(3), 244–258. https://doi.org/10.1016/j.tra.2005.07.001

Daamen, W., Goverde, R. M. P., & Hansen, I. A. (2009). Non-discriminatory automatic registration of knock-on train delays. *Networks and Spatial Economics*, *9*(1), 47–61. https://doi.org/10.1007/s11067-008-9087-2

Daamen, W., Houben, T., Goverde, R., Hansen, I., & Weeda, A. (2006). Monitoring system for reliability of rail transport chains.

Delay Attribution Board. Delay Attribution Guide (2017). he United Kingdom of Great Britain and Northern Ireland. Retrieved from http://www.delayattributionboard.co.uk/documents/dag_pdac/April 2017 Delay Attribution Guide.pdf

Deng, X., Liu, Q., Deng, Y., & Mahadevan, S. (2016). An improved method to construct basic probability assignment based on the confusion matrix for classification problem. *Information Sciences*, *340–341*, 250–261. https://doi.org/10.1016/j.ins.2016.01.033

Evans, G. W., Wener, R. E., & Phillips, D. (2002). The Morning Rush Hour: Predictability and Commuter Stress. *Environment*, *34*(4), 521–530.

Fioole, P.-J. (2018a). Personal Interview.

Fioole, P.-J. (2018b). Personal observation of Accuracy of train departure delay prediction 20 minutes to the future.

Fioole, P.-J., & Tielman, W. (2019). Personal interview.

Galappaththi, A. (2015). Overfitting. In *Young Researchers' Forum – PGIS* (Vol. 2, pp. 60–61).

Gatersleben, B., & Uzzell, D. (2007). Affective appraisals of the daily commute: Comparing perceptions of drivers, cyclists, walkers, and users of public transport. *Environment and Behavior*, *39*(3), 416–431. https://doi.org/10.1177/0013916506294032

Gibson, S., Cooper, G., & Ball, B. (2002). Developments in transport policy: The evolution of capacity charges on the UK rail network. *Journal of Transport Economics and Policy*, *36*(2), 341–354.

Good, P. I. (2006). *Resampling Methods* (Third). Boston, United States of America: Birkhäuser.

Goverde, R. M. P. (2010). A delay propagation algorithm for large-scale railway traffic networks. *Transportation Research Part C: Emerging Technologies*, *18*(3), 269–287. https://doi.org/10.1016/j.trc.2010.01.002

Goverde, R. M. P., Daamen, W., & Hansen, I. A. (2008). Automatic identification of route conflict occurrences and their consequences. *Computers in Railways XI*, *103*(March 2016), 473–482. https://doi.org/10.2495/CR080461

Gudden, J. (2014). *Does the predictability of the commute mediates the relation of commuting mode on stress ?* Tilburg University.

Hansen, I. A., Goverde, R. M. P., & Van Der Meer, D. J. (2010). Online train delay recognition and running time prediction. In *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*. https://doi.org/10.1109/ITSC.2010.5625081

Harrod, S., Cerreto, F., & Nielsen, O. A. (2019). A closed form railway line delay propagation model. *Transportation Research Part C: Emerging Technologies*, *102*(February), 189–209. https://doi.org/10.1016/j.trc.2019.02.022

Hawkins, D. M. (2004). The Problem of Overfitting. *Journal of Chemical Information and Computer Sciences*, *44*(1), 1–12. https://doi.org/10.1021/ci0342472

He, T. (2015). Kaggle Winning Solution Xgboost algorithm -- Let us learn from its author. Retrieved July 4, 2019, from https://www.slideshare.net/ShangxuanZhang/kaggle-winning-solution-xgboost-algorithm-let-us-learn-from-its-author

Hellsten, E., Haahr, J. T., & Van der Hurk, E. (2018). *Train Delay Prediction in the Netherlands through Neural Networks*. Lyngby. Retrieved from http://orbit.dtu.dk/en/publications/train-delay-prediction-in-the-netherlands-through-neural-networks(023f031f-921f-495e-91b0-53e9ea14381d).html

Hesterberg, T. C. (2015). What Teachers Should Know About the Bootstrap: Resampling in the Undergraduate Statistics Curriculum. *American Statistician*, *69*(4), 371–386. https://doi.org/10.1080/00031305.2015.1089789

Huisman, T., & Boucherie, R. J. (2001). Running times on railway sections with heterogeneous train traffic. *Transportation Research Part B: Methodological*, *35*(3), 271–292. https://doi.org/10.1016/S0191-2615(99)00051-X

Husdal, J. (2004). Reliability and Vulnerability versus Cost and Benefits, Molde University College. In *Proceedings of the second International Symposium on Transportation Network Reliability (INSTR)* (pp. 180–186). Christchurch and Queenstown.

INFORMS. (2018a). Current Competition Data. Retrieved December 18, 2018, from https://connect.informs.org/railway-applications/awards/problem-solving-competition/new-item2

INFORMS. (2018b). Current Competition Result. Retrieved December 18, 2018, from https://connect.informs.org/railway-applications/awards/problem-solving-competition/result

INFORMS. (2018c). Detailed problem description and instructions. Retrieved December 18, 2018, from https://connect.informs.org/railway-applications/awards/problem-solving-competition/new-item2

INFORMS. (2018d). Problem Benchmark Method. Retrieved December 18, 2018, from https://connect.informs.org/railway-applications/awards/problem-solving-competition/new-item2

INFORMS. (2018e). RAS Problem Solving Competition. Retrieved December 18, 2018, from https://connect.informs.org/railway-applications/awards/problem-solving-competition

INFORMS. (2019a). 2018 Archive. Retrieved from https://connect.informs.org/railway-applications/new-item3/problem-solving-competition681/new-item12

INFORMS. (2019b). RAS Problem Solving Competition. Retrieved from https://connect.informs.org/railway-applications/new-item3/problem-solving-competition681

Jain, A. (2016). Complete Guide to Parameter Tuning in XGBoost with codes in Python. Retrieved July 4, 2019, from https://www.analyticsvidhya.com/blog/2016/03/complete-guide-parameter-tuning-xgboost-with-codes-python/

Jain, M. (2018). Hyperparameter tuning in XGBoost using genetic algorithm. Retrieved July 4, 2019, from https://towardsdatascience.com/hyperparameter-tuning-in-xgboost-using-genetic-algorithm-17bd2e581b17

Jaiswal, S., Bunker, J., & Ferreira, L. (2009). Modelling the Relationships Between Passenger Demand and Bus Delays at Busway Stations. *Transportation Research Board 88th Annual Meeting*, (January), 11–15.

Jerome H. Friedman. (2001). Greedy Function Approximation: A Gradient Boosting Machine. *Annals of Statistics*, 1189–1232.

Kaggle Participant. (2017). Hyperparameter Grid Search with XGBoost. Retrieved from https://www.kaggle.com/tilii7/hyperparameter-grid-search-with-xgboost

Kecman, P. (2014). *Models for Predictive Railway Traffic Management*. Delft University of Technology, Delft. https://doi.org/10.4233/uuid:539e96b3-18d7-4f6f-b662-ce22ae269f2a

Kecman, P., Corman, F., & Meng, L. (2015). Train delay evolution as a stochastic process. In *6th International Conference on Railway Operations Modelling and Analysis - RailTokyo2015*. Tokyo.

Kecman, P., & Goverde, R. M. P. (2015a). Online Data-Driven Adaptive Prediction of Train Event Times. *IEEE Transactions on Intelligent Transportation Systems*, *16*(1), 465–474. https://doi.org/10.1109/TITS.2014.2347136

Kecman, P., & Goverde, R. M. P. (2015b). Predictive modelling of running and dwell times in railway traffic. *Public Transport*, *7*(3), 295–319. https://doi.org/10.1007/s12469-015-0106-7

Kesar Singh and Minge Xie. (n.d.). *Bootstrap: A Statistical Method*.

KNMI. (n.d.-a). About KNMI. Retrieved September 5, 2019, from https://www.knmi.nl/over-het-knmi/about

KNMI. (n.d.-b). Daggegevens van het weer in Nederland. Retrieved from https://www.knmi.nl/nederland-nu/klimatologie/daggegevens

KNMI. (n.d.-c). Klimatologie, Uurgegevens van het weer in Nederland – Download. Retrieved January 16, 2019, from https://projects.knmi.nl/klimatologie/uurgegevens/selectie.cgi

Lam, W. H. K., Cheung, C. Y., & Lam, C. F. (1999). A study of crowding effects at the Hong Kong light rail transit stations. *Transportation Research Part A: Policy and Practice*, *33*(5), 401–415. https://doi.org/10.1016/S0965-8564(98)00050-0

Lee, W. H., Yen, L. H., & Chou, C. M. (2016). A delay root cause discovery and timetable adjustment model for enhancing the punctuality of railway services. *Transportation Research Part C: Emerging Technologies*, *73*, 49–64. https://doi.org/10.1016/j.trc.2016.10.009

Lessan, J., Fu, L., & Wen, C. (2019). A hybrid Bayesian network model for predicting delays in train operations. *Computers and Industrial Engineering*, *127*, 1214–1222. https://doi.org/10.1016/j.cie.2018.03.017

Li, Z., Hensher, D. A., & Rose, J. M. (2010). Willingness to pay for travel time reliability in passenger transport: A review and some new empirical evidence. *Transportation Research Part E: Logistics and Transportation Review*, *46*(3), 384–403. https://doi.org/10.1016/j.tre.2009.12.005

Lindfeldt, A. (2010). A study of the performance and utilization of the Swedish railway network. *First International Conference on Road and Rail Infrastructure - CETRA2010*. Retrieved from https://www.kth.se/polopoly_fs/1.160196!/Menu/general/column-content/attachment/Lindfeldt_2010_Performance_and_Utilization.pdf

Ling, X., Peng, Y., Sun, S., Li, P., & Wang, P. (2018). Uncovering correlation between train delay and train exposure to bad weather. *Physica A: Statistical Mechanics and Its Applications*, *512*, 1152–1159. https://doi.org/10.1016/j.physa.2018.07.057

Luo, H., Xu, J., Wu, Q., & Gao, Z. (2018). *A Railway Delay Prediction Model Based On Non-Homogeneous Markov Chains*.

Machine Learning Mastery. (2019). How to Evaluate Gradient Boosting Models with XGBoost in Python. Retrieved September 8, 2019, from https://machinelearningmastery.com/evaluate-gradient-boosting-models-xgboost-python/

Maimon, O., & Rokach, L. (2005). *Data Mining and Knowledge Discovery Handbook*. (O. Maimon & L. Rokach, Eds.). https://doi.org/10.1007/978-0-387-69935-6_2

Mannhardt, F., & Landmark, A. D. (2019). Mining railway traffic control logs. *Transportation Research Procedia*, *37*, 227–234. https://doi.org/10.1016/j.trpro.2018.12.187

Marković, N., Milinković, S., Tikhonov, K. S., & Schonfeld, P. (2015). Analyzing passenger train arrival delays with support vector regression. *Transportation Research Part C: Emerging Technologies*, *56*, 251–262. https://doi.org/10.1016/j.trc.2015.04.004

Middelkoop, A. D. (2010). Headway generation with ROBERTO. *WIT Transactions on the Built Environment*, *114*, 431–439. https://doi.org/10.2495/CR100401

Mobiliteitsbeeld 2017. (2017). Retrieved from https://www.rijksoverheid.nl/documenten/rapporten/2017/10/23/kim-publicatie-mobiliteitsbeeld-2017

Murray, A. T., & Grubesic, T. (2007). *Critical infrastructure: Reliability and vulnerability*. *Springer Science & Business Media*. https://doi.org/10.1007/978-3-642-60714-1

Nabian, M. A., Alemazkoor, N., & Meidani, H. (2018). *Predicting Near-Term Train Schedule Performance and Delay Using Bi-Level Random Forests*.

Nagpal, A. (2017). L1 and L2 Regularization Methods. Retrieved July 4, 2019, from https://towardsdatascience.com/l1-and-l2-regularization-methods-ce25e7fc831c

Nair, R., Hoang, T. L., Laumanns, M., Chen, B., Cogill, R., Szabó, J., & Walter, T. (2019). An ensemble prediction model for train delays. *Transportation Research Part C: Emerging Technologies*, *104*(April), 196–209. https://doi.org/10.1016/j.trc.2019.04.026

Nargesian, F., Samulowitz, H., Khurana, U., Khalil, E. B., & Turaga, D. (2017). Learning feature engineering for classification. *IJCAI International Joint Conference on Artificial Intelligence*, (August), 2529–2535. https://doi.org/10.24963/ijcai.2017/352

Neves, D. V. A. (2017). *Impacts of Winter Related Railway Disruption on Network Performance*. University of Twente.

Ng, A. Y. (2004). Feature selection, L 1 vs. L 2 regularization, and rotational invariance. In *Twenty-first international conference on Machine learning - ICML '04* (p. 78). https://doi.org/10.1145/1015330.1015435

NS. (2016). Vervoerplan 2017. Utrecht. Retrieved from https://www.ns.nl/binaries/_ht_1533726797181/content/assets/ns-nl/over-ons/vervoerplan-2017.pdf

NS. (2017a). Intercity (DDZ). Utrecht. Retrieved from https://www.ns.nl/binaries/_ht_1502695331160/content/assets/ns-en/about-ns/2017/ddz.pdf

NS. (2017b). Sprinter (SLT). Utrecht. Retrieved from https://www.ns.nl/binaries/_ht_1502695322484/content/assets/ns-en/about-ns/2017/slt.pdf

NV Nederlandse Spoorwegen. (2019). *NS Annual Report 2018*. Retrieved from https://www.nsjaarverslag.nl/

Olsson, N. O. E., & Haugland, H. (2004). Influencing factors on train punctuality - Results from some Norwegian studies. *Transport Policy*, *11*, 387–397. https://doi.org/10.1016/j.tranpol.2004.07.001

Oneto, L., Fumeo, E., Clerico, G., Canepa, R., Papa, F., Dambra, C., … Anguita, D. (2016). Advanced analytics for train delay prediction systems by including exogenous weather data. In *IEEE International Conference on Data Science and Advanced Analytics* (pp. 458–467). https://doi.org/10.1109/DSAA.2016.57

Oneto, L., Fumeo, E., Clerico, G., Canepa, R., Papa, F., Dambra, C., … Anguita, D. (2017). Dynamic Delay Predictions for Large-Scale Railway Networks: Deep and Shallow Extreme Learning Machines Tuned via Thresholdout. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, *47*(10), 2754–2767. https://doi.org/10.1109/TSMC.2017.2693209

Oneto, L., Fumeo, E., Clerico, G., Canepa, R., Papa, F., Dambra, C., … Anguita, D. (2018). Train Delay Prediction Systems: A Big Data Analytics Perspective. *Big Data Research*, *11*, 54–64. https://doi.org/10.1016/j.bdr.2017.05.002

Palmqvist, C. W., Olsson, N. O. E., & Hiselius, L. W. (2017). Some influencing factors for passenger train punctuality in Sweden. *International Journal of Prognostics and Health Management*, *8*(Special Issue 7), 0–13.

Pek, J., Wong, A. C. M., & Wong, O. C. Y. (2017). Confidence Intervals for the Mean of Non-Normal Distribution: Transform or Not to Transform. *Open Journal of Statistics*, *07*(03), 405–421. https://doi.org/10.4236/ojs.2017.73029

Peters, J., Emig, B., Jung, M., & Schmidt, S. (2006). Prediction of Delays in Public Transportation using Neural Networks, 92–97. https://doi.org/10.1109/cimca.2005.1631451

Restrepo, M. (2018). Doing XGBoost hyper-parameter tuning the smart way — Part 1 of 2. Retrieved July 4, 2019, from https://towardsdatascience.com/doing-xgboost-hyper-parameter-tuning-the-smart-way-part-1-of-2-f6d255a45dde

Rokach, L., & Maimon, O. (2015). *Data Mining With Decision Trees*. (H. Bunke & P. S. P. Wang, Eds.) (2nd ed.). Singapore: World Scientific Publishing Co. Pte. Ltd. 5.

San, H. P., & Mohd Masirin, M. I. (2016). Train dwell time models for rail passenger service. *MATEC Web of Conferences*, *47*. https://doi.org/10.1051/matecconf/20164703005

Schapire, R. E. (2013). Explaining AdaBoost. In *Empirical Inference: Festschrift in Honor of Vladimir N. Vapnik* (pp. 37–52). Springer. https://doi.org/10.1007/978-3-642-41136-6

SciPy community. (2019). scipy.stats.kurtosistest. Retrieved August 5, 2019, from https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.kurtosistest.html#scipy.stats.kurtosistest

Smola, A., & Vishwanathan, S. V. N. (2008). *Introduction to Machine Learning*. Cambridge University, United Kingdom.

Sokolova, M., & Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. *Information Processing and Management*, *45*(4), 427–437. https://doi.org/10.1016/j.ipm.2009.03.002

Sposato, R. G., Röderer, K., & Cervinka, R. (2012). The influence of control and related variables on commuting stress. *Transportation Research Part F: Traffic Psychology and Behaviour*, *15*, 581–587. https://doi.org/10.1016/j.trf.2012.05.003

TAP-TSI. (2019). EG2 - Delay Reason Code - TAP-TSI - UIC. Retrieved November 14, 2019, from https://tap-tsi.uic.org/IMG/xls/annex_10_taf_tap_coding_list_20120511.xls

Tirachini, A., Hensher, D. A., & Rose, J. M. (2013). Crowding in public transport systems: Effects on users, operation and implications for the estimation of demand. *Transportation Research Part A: Policy and Practice*, *53*, 36–52. https://doi.org/10.1016/j.tra.2013.06.005

Van den Bulk, L., Fioole, P., & Kachergis, G. (2018). *Predicting Short Term Train Delays in the Dutch Rail Network*. Utrecht.

Van der Hurk, E. (2019). Personal interview. Utrecht.

Van Hagen, M., & De Bruyn, M. (2012). The Ten Commandments of How To Become a Customer-Driven Railway Operator. In *European transport Conference* (pp. 1–19). Glasgow.

Wang, R., & Work, D. B. (2015). Data Driven Approaches for Passenger Train Delay Estimation. In *2015 IEEE 18th*

International Conference on Intelligent Transportation Systems. Las Palmas, Spain. https://doi.org/10.1109/ITSC.2015.94

Whitley, D. (Colorado S. U. (1994). A Genetic Algorithm Tutorial. *Statistics and Computing*, (4), 65–85. Retrieved from http://samizdat.mines.edu/ga_tutorial/

Wikipedia. (2019). Earth radius. Retrieved from https://en.wikipedia.org/wiki/Earth_radius

Willmott, C., & Matsuura, K. (2005). Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. *Climate Research*, *30*, 79–82. Retrieved from www.int-res.com

XGBoost Developers. (n.d.). XGBoost Documentation. Retrieved January 22, 2019, from https://xgboost.readthedocs.io/en/latest/index.html

XGBoost Developers. (2016a). Contribute to XGBoost. Retrieved January 22, 2019, from https://xgboost.readthedocs.io/en/latest/contribute.html

XGBoost Developers. (2016b). XGBoost Parameters. Retrieved from https://xgboost.readthedocs.io/en/latest/parameter.html

XGBoost Developers. (2016c). XGBoost Python API Reference. Retrieved from https://xgboost.readthedocs.io/en/latest/python/python_api.html

Xia, Y., Liu, C., Li, Y. Y., & Liu, N. (2017). A boosted decision tree approach using Bayesian hyper-parameter optimization for credit scoring. *Expert Systems with Applications*, *78*, 225–241. https://doi.org/10.1016/j.eswa.2017.02.017

Xia, Y., Van Ommeren, J. N., Rietveld, P., & Verhagen, W. (2013). Railway infrastructure disturbances and train operator performance: The role of weather. *Transportation Research Part D: Transport and Environment*, *18*(1), 97–102. https://doi.org/10.1016/j.trd.2012.09.008

Yaghini, M., Khoshraftar, M., & Seyedabadi, M. (2013). Railway passenger train delay prediction via neural network model. *Journal of Advanced Transportation*, *47*, 355–368. https://doi.org/10.1002/atr

Yazdani, D., Omidvar, M. N., Deplano, I., Lersteau, C., Makki, A., Wang, J., & Nguyen, T. T. (2019). Real-time seat allocation for minimizing boarding/alighting time and improving quality of service and safety for passengers. *Transportation Research Part C: Emerging Technologies*, *103*(April), 158–173. https://doi.org/10.1016/j.trc.2019.03.014

Yuan, J. (2007). Dealing with Stochastic Dependence in the Modeling of Train Delays and Delay Propagation. In *International Conference on Transportation Engineering 2007* (pp. 3908–3914). Reston, VA: American Society of Civil Engineers. https://doi.org/10.1061/40932(246)641

Yuan, J., Goverde, R. M. P., & Hansen, I. a. (2002). Propagation of train delays in stations. *Computers in Railways Viii*, *13*, 975–984\n1163.

Zhou, L., Pan, S., Wang, J., & Vasilakos, A. V. (2017). Machine learning on big data: Opportunities and challenges. *Neurocomputing*, *237*, 350–361. https://doi.org/10.1016/j.neucom.2017.01.026

Zieger, S., Weik, N., & Nießen, N. (2018). The influence of buffer time distributions in delay propagation modelling of railway networks. *Journal of Rail Transport Planning and Management*, *8*, 220–232. https://doi.org/10.1016/j.jrtpm.2018.09.001

Appendix

A. Data

A.1. Timetable data

Dataset glossary

Table 40 Planned timetable dataset glossary

Name	Type and unit	Description
Day	Date	The traffic date. Date changes at 02:00 am.
Train number	Integer	An identification number of a train assigned to the planned train activity.
Direction	String	An indication of a direction: E (even) and O (odd). It refers to the way train numbers are assigned (odd and even for each direction).
Location	String	The abbreviation of the station or timetable control point (the points in between stations where the trains are planned on and measured on).
Abroad	Boolean	Specifies whether the location is abroad. J = Yes (Ja), N = No (Nee).
Activity	String	The activity of the train at the location at the time. V = Departure (Vertrek), D = Pass through (Doorgaan), A = Arrival (Aankomst). K_A and K_V denote short (Korte) stop where arrival and departure are planned within one minute.
Order number	Integer	An order number of the activity of the train.
Planned Time	Date Time	The planned date and time for the activity.
Train Characteristic	String	The train characteristic. IC = Intercity, SP or ST = Sprinter, LM = Empty train
Pattern	String	The first character indicates the timetable hour pattern and the second part indicates the train series.
Bare Driving Time	Integer [seconds]	The calculated minimum driving time from the preceding to the current location.
Traction Type	String	Type of planned rolling stock.
Timetable Speed	Integer [km/h]	The maximum allowed speed according to the timetable.

Data processing

A column identifying the train series number was added. The train series number was retrieved as the numeric part of the string in the column Pattern. Where there was no numeric part of the Pattern, the train series number was retrieved from the train number by subtracting the value of tens and ones. A list of unique values of train series was stored as a reference list of train series occurring in the timetable. Next, the direction was changed from O and E to 0 and 1 as numeric values are easier to process than strings. Other than that, no major corrections nor adjustments were necessary besides dropping unused columns.

A.2. Realization data

Dataset glossary

Table 41 Realization dataset Glossary

Name	Type and unit	Description
Traffic Date	Date	The traffic date.
Train series	String	The train series and direction.
Train Characteristic	Factor	Same as Train Characteristic in Table 7 Planned timetable dataset example.
Train number	Integer	Same as Train number in Table 7 Planned timetable dataset example.
Location	Factor	Same as Location in Table 7 Planned timetable dataset example.
Activity	Factor	Same as Activity in Table 7 Planned timetable dataset example.
Planned Time	Date Time	Same as Planned Time in Table 7 Planned timetable dataset example.
Realization	Date Time	The time when the train completed the corresponding activity.
Delay	Integer [minutes]	The delay of the train at the activity. Rounded to whole minutes.
Delay Jump	Integer [minutes]	The change of the train's delay from the previous activity. Rounded to whole minutes.
Cause	String	Possible cause of the delay jump if known. Aggregated into three categories: *Other train* (Delay jump due to a knock-on effect), *Other primary cause* (Delay jump due to a primary cause) and *Stopping procedure* (Delayed departure). If empty, the cause is either irrelevant (driver behavior) or unknown.

Data processing

Realization dataset is fundamental for our work as it mimics online collected data that would be used for a delay prediction in an application in reality. It contains an information about the train's identification, its actual location, respectively the last point of registration, and the delay at that point. All the other data from the further datasets are fitted to it based on the train identification, spatial and/or temporal relevance. The target information from this dataset was the train identification (the train series and the train number), the train type ('characteristic') and the event information: the location, the activity, and the planned and realized time. The train series identification was split into the numerical and letter type denoting the train series number and the direction. It is a common practice of the dispatchers in the Netherlands to add 300,000 to the train (series) number to virtually create a new train when a train is for example cancelled and rescheduled, turned around before reaching its terminal station, or any other major intervention is applied. It is done to avoid complications and confusion such major rescheduling could lead to as for example seemingly large (negative) delay. 300,000 was therefore subtracted from such trains (series) numbers.

Train types were initially IC (intercity), SPR (sprinter), ST (stopping train), LM (empty rolling stock), LL (empty rolling stock) and NTT (not relevant). Due to similarity in characteristics between ST and SPR, ST was reclassified as SPR (and will be considered as a part of this class from now on), and LL and NTT were reclassified as LM which can therefore be understood as other than IC or SPR. Furthermore, trains of train series of 800,000 or higher were assigned the LM train type as those trains are supposed to belong to that category despite being registered as IC or SPR in the realization data.

The delay was recalculated as a difference between the planned and realized time and was rounded up to quarters of a minute to increase precision of the information yet keeping it moderately concise as the aspired precision of the delay predictions is in minutes, not in seconds. The registered delays were checked for illogical and missing values. A negative delay of IC and SPR trains of more than one minute on a departure (meaning an early departure) was considered illogical as such early departures are strictly not allowed. Early passage through intermediate locations and arrivals were considered illogical if they exceeded 10 minutes. Negative delays exceeding the thresholds were marked as missing values which was the case of 0.007% entries. Similarly, delays exceeding an upper limit which was set to 45 minutes were marked as a missing value which occurred in 0.008% entries. It is highly unlikely that such large delay would not be addressed by the dispatchers. Possible explanation of presence of early activities is that they were caused by dispatching interventions and confusion of train series identification.

Seemingly unrealistic high changes in delay in between consecutive activities were examined. In multiple cases where there was a sudden increase in a delay of up to dozens of minutes, a source of the delay was always registered, and the delay jump therefore explained. Unrealistically large negative delay jumps did not occur.

Logical time flow of the train registrations was checked for errors meaning that every registration of a train had to be later than its registration at the preceding registration point. This constraint was violated in the order of tens cases which were marked as a missing value and therefore disregarded in the delay prediction process.

A.3. Interactions data

Dataset glossary

Table 42 Minimum required headways dataset glossary

Name	Type and unit	Description
Train 1 series	Integer	Train series number of the leading train. Unspecified direction.
Train 2 series	Integer	Train series number of the following train. Unspecified direction.
Train 1 type	String	Rolling stock type of the leading train.
Train 2 type	String	Rolling stock type of the following train.
Area name	String	Code of the timetable point where the signal is located.
Signal name	String	The signal number on which the headway time is calculated.
Local min headway	Double [s]	The local minimum headway time according to the tool at the signal.

Data processing

For the purpose of infrastructure assignment, the direction in which a train approaches an infrastructure point (signal) matters. The direction of trains with respect to the timetable, which is denoted as E (even) or O (odd) is in the context of minimum required headways irrelevant as a train running under an even number in one series might drive through a signal in the same direction as another train running under a an odd number of another train series.

As the train interactions and minimum required headways were to be assigned to relevant timetable instances, it was necessary to identify train series directions. This was possible only when a sequence of at least two rows in an uninterrupted block was present. The sequence of the Area names reading in rows from top to down was then looked for in the timetable of the relevant train series. Existence of multiple signals per location was ignored and only unique row for an Area name was considered for observing the direction to match the level of detail of the timetable which does not contain any information about the signals. The identified direction was then assigned to all relevant rows for the location.

Many of the rows in the dataset were duplicates but as the order of the rows was important to determine the directions, duplicate rows had to be kept for the direction identification step. In the first step of direction identification, the direction was successfully identified only for less than a half of the rows (approximately 46%). Some of the locations within the route in the headway's dataset might not have been in the timetable due to not being a timetable point or due to a difference in locations names used by NS and ProRail although the rest of the route or its parts were successfully identified. Such rows without an identified direction were assigned with a direction assigned to rows belonging to the relevant block of rows. Understand a block of rows as a sequence of rows directly following each other where the leading and following train series couple is unchanged. Although this assumption is not 100% true (there could be headways for the same pair of train series in the opposing directions following each other), it is expected to be sufficient as all possible train series pairs sharing the part of infrastructure in the given direction are listed and only then the other direction should follow.

After removing duplicate rows, the second step lead to more than 68% of the rows being assigned with a direction. Next, not all train series were relevant as not all of them appeared in the realization data and the planned timetable and could not be coupled with the realization data later. Therefore, only couples with both series being relevant were kept. Furthermore, the data were aggregated on the Area name level per sequence of interactions by a couple of train series and a direction. As the minimum required headway varied largely among the locations within each Area, the maximum value was kept as the most critical in delay propagation. The minimum required headway was then converted to minutes and rounded up with a precision of 0.5 minute. At this point, approximately 85% of rows were successfully assigned with a direction for both trains. Eventually, remaining rows were disregarded as there was no possibility to reliably determine the direction. This mainly concerns trains sharing a single infrastructure point and therefore not showing a sequence of locations from which a direction could be determined.

The following step was to identify the actual train interactions. To do so, patterns of train interactions were created from the timetable which were later used for interaction identification in the realization data. This is a reason that the train interactions are referred to as off-line-identified and it means that only train interactions included in the timetable are considered. Any interactions due to operational changes are not foreseen.

The interaction patterns based on the timetable has a form of a matrix where the rows refer to the rows of the timetable and the columns refer to all the trains series that might cause any interaction with any other train.

For each timetable row, a list of locations the train will pass in the upcoming 20 minutes was made. In the columns, an interaction would be identified if there was any minimum required headway at any of the upcoming locations with the train series number relevant to the column.

The identified interactions were stored in multiple matrices: binary identification of existence of an interaction, the minimum required headway at the first shared location, direction of the leading train and planned time of the leading train passing the first shared location. From the planned time and the minimum required headway, planned time margins were calculated. Margins exceeding 15 minutes were considered large enough to have low impact on delay propagation as an intervention from the dispatching would be put in place in such case. Margins bellow 0 were understood as that there is no interaction between the trains. In these two cases, the interaction was disregarded. This step was necessary as there was no control of the actual existence of an interaction at a specific time: interactions were identified on trains series level, not train number level.

A.4. Passenger data

Dataset glossary

Table 43 Passenger counts dataset glossary

Name	Type	Description
Date	Date	The date.
Train number	Integer	Train number.
Departure station	String	The departure station.
Departure time	Date Time	The time of departure from the station.
Arrival station	String	The arrival station.
Arrival time	Date Time	The time of arrival to the station.
Total number of passengers	Double	The computed number of passengers travelling from the departure to the arrival station by this train.
Number of boarding passengers	Double	The computed number of passengers boarding the train in the departure station.
Planned seats	Integer	The planned number of seats on this train.
Realized seats	Integer	The realized number of seats on this train.

Data processing

According to policy of NS regarding a seating capacity of the trains described in ‘Vervoerplan 2017’ (translated as Transport plan 2017), passengers with a first-class ticket should always find an empty seat as well as passengers traveling in the second class during off-peak hours and during the weekends. During the peak hours on working days, passengers in the second class should find an available seat on Intercity trains if their journey by the train exceeds 15 minutes. Otherwise, including trips by the Sprinter trains, there should be enough space for the passengers to stand. (NS, 2016)

NS operates with a number of rolling stock types with varying characteristics such as driving performance related to acceleration and braking. A simplification was made…….. As a reference for processing of passenger data, two rolling stock types are considered: DDZ for Intercity trains and SLT for Sprinter trains. The two rolling stock types were used as a reference for assessment of the maximum possible capacity provided according to the passenger data. That was necessary because the extreme were reaching a maximum planned capacity of 2,388 seats while the utmost highest possible capacity of an IC train is 1214 seats. A limit of possible planned capacity was therefore set to 1214 seats for IC trains (referring to the DDZ rolling stock in its largest coupling) and 1005 seats for SPR trains (considering SLT rolling stock in its largest coupling). Furthermore, unrealistic numbers of passengers were filtered out by introducing a limit equal to 1.6 times the seat capacity of IC trains and 2.4 times capacity of SPR trains. The coefficients refer to the maximum capacity calculated from online brochures about the rolling stock publicized by NS (2017a, 2017c). In total, 213 entries of the dataset were removed, most of them belonging to IC trains.

There is no information about what rolling stock was used for each trip which might have differed to what was scheduled in the timetable for the respective trips. Based on the number of seats provided, the rolling stock type theoretically could be derived. The risk of introducing errors in the data by doing so was found to be too high in comparison to the potential benefits. Above that, it was the difference between planned and realized capacity that was targeted for our research. Furthermore, if there is no difference between the planned and realized capacity, it implies that the planned rolling stock was used and the possible effects of the rolling stock on the systems behavior are likely implicit in the train series number, the day and the time as it is a characteristic which is being regular for that train series at that time. On the other hand, a difference in the seating capacity identifies a difference in the rolling stock comparing what was planned and what was realized. Creating an additional feature identifying a difference in rolling stock would only be a duplicate information.

As the model is trained for each train series separately, introducing a feature of rolling stock type would make sense only if more than two rolling stock types were used for trains belonging to the respective train series and that only if each of the rolling stock types was represented a sufficient number of times which in general is not the case. Introduction of the rolling stock type into the feature sets would essentially result in creating a category 'regular rolling stock type' and 'irregular rolling stock type' which would be translated as 0 and 1 in a machine learning language. If *n* rolling stock types were considered, those would be translated to 0, 1…n-1. As a rolling stock type coded as for example 3 is not necessarily any greater than a rolling stock type coded as 1, such coding would be missing any weighting changing the categorical variable into ordinal. That brings us back to the ratio of planned and realized seating capacity which is used as an identification of a possible irregular situation in passenger movements.

The value lies on the interval between 0 (a cancelled train) and infinity (an unplanned additional train which was observed in approximately 13 000 entries). Omitting the unplanned trains, the highest ratio of planned and realized seats was 4.39 and was greater than 1 in nearly 45,000 cases or greater than 1.5 in approximately 5,000 cases. If this feature has a value of 1, it means the planned rolling stock type was used as the seating capacity is identical (there are no pairs of different rolling stock types with the same seating capacity). All values different from 1 identify a usage of an irregular rolling stock type. Seating capacity therefore can be seen as transformation of a categorical variable of rolling stock types into an ordinal variable which is easier interpretable in machine learning language. The reason for using a ratio of realized and planned seating capacity instead of using them as two separate features, or using just one of them, is to enhance the drop or increase in capacity which may lead to more or less crowding than was accounted for.

In the case the machine learning model is trained for each train series separately, the ratio of planned and realized seats therefore carries more information than introducing a feature representing the rolling stock type used.

Besides some cases of an exceeded capacity, a few cases of extremely high number of boarding passengers appeared. The highest number of passengers boarded was 1880 followed by set of similarly high numbers. Although the total capacity limit was not exceeded, it indicated a need to check reasonability of those entries. The locations with the highest numbers of boarding passengers were controlled and turned out to be Utrecht, multiple stations of Amsterdam, Den Haag, Rotterdam and Leiden followed by other stations. As these indeed are busy locations and no unreasonable location was found in the entries with a high number of boarding or alighting passengers, no entries were filtered out due to a high number of boarding or alighting passengers, neither as outliers because it is the extreme numbers of passengers that might affect delays development and therefore it is not desired to filter such data out if unnecessary. However, 6 entries were removed as the number of boarding passengers exceeded the total number of passengers leaving on the train from the location which would mean some of the passengers got on the train and immediately left it again. In 5 of those cases, it occurred at the first station of the train.

The date, train number and location were used for coupling with the realization data. The main information of this dataset lies in the numbers of passengers and the seating capacity. The total number of passengers refers to the number of passengers travelling between the two locations, meaning it includes passengers who arrived at the departure location minus those who left the train there, plus passengers that boarded at the departure location. The number of alighting passengers at a location was therefore derived from the total number of passengers that left from the preceding location, the number of passengers that boarded the train at the current location, and the total number of passengers that departed from the there. The numbers of passengers were given in numbers with precision of 8 decimal points. Assuming that only whole passengers can travel by train, the numbers were eventually rounded to integers. Nevertheless, due to calculations with the float numbers, minor errors were introduced, and negative numbers of alighting passengers were observed. Vast majority of those, approximately 1.87% of the entries, lied between 0 and -1 and was considered as an error induced by the calculations and was assumed to be equal to 0. In approximately 0.06% of the entries, there was a missing entry in the preceding departure which lead to a high negative number of alighting passengers. The number of alighting passengers was fixed to -1, representing a missing value. In total, the number of alighting passengers was set to -1 to represent a missing value in nearly 0.43% of the final dataset contained a missing value.

Realization data contain train registrations at all timetable locations, including locations where there are no passenger movements and the train only passes through. Furthermore, there are two entries for each station, an arrival and a departure (except for the first and the terminal locations naturally). In contrary to that, passenger data entries are for pairs of locations. Therefore, a strategy to couple the passenger data with the realization data had to be made. Although the currently applied passenger data were generated retrospectively, it is assumed that they were collected in real time. The final numbers of passengers boarding and alighting as a station can be registered when the doors of the train close and the train departs. It was therefore decided to pair the passenger data with the realization data on the departures. The numbers of passengers do not change until the next stop and are therefore copied across the entire upcoming section of that specific train until the next departure.

A.5. Weather data

Dataset glossary

Table 44 Weather dataset glossary

Name	Type and unit	Description
Station number	Integer	An identification number of the station.
YYYYMMDD	Date	A date of the measurement.
Hour	Integer	An hour of the measurement.
Average wind speed	Integer [0.1 m/s]	
Highest wind speed	Integer [0.1 m/s]	
Temperature	Integer [0.1 °C]	
Min temperature	Integer [0.1 °C]	
Precipitation	Integer [0.1mm]	
Horizontal view	Integer [scaled]	
Mist	Boolean (Present or not)	
Rain	Boolean (Present or not)	
Snow	Boolean (Present or not)	
Storm	Boolean (Present or not)	
Ice	Boolean (Present or not)	

Data processing

The data file downloaded from the website of KNMI was in a ‘*.txt’ format. The file contained a header, information about the weather stations and then the data followed. The weather stations’ names and coordinates were extracted into one file (see Table 45), and the weather data were saved as another separate file (see Table 18). To couple the weather data with time- and space-vice corresponding realization data, each of the entries in the realization data had to be identified by the ‘realization’ time, specifically by the year, month, day and an hour when the train was registered. As the realization data were in a 00-23-hour format while the weather data were in a 01-24-hour format (because the measurement refers to the past hour), one hour had to be added to the hour value in the realization data. Time and date of the weather data were formatted in the same way.

Unfortunately, not all quantities were available for all the stations nor for all the time stamps. Stations or quantities with a significant amount of missing data had to be removed from the dataset to ensure relevant and reliable information. To completely avoid issues with missing data, stations with any missing data were removed. Still, sufficient number of weather stations was left and importantly, they still were evenly distributed across the country. However, not all the remaining stations provided data for all the quantities. Two groups of weather stations were therefore created, each providing data for all the time stamps within the required time range in a subset of the quantities that were available. See Table 49 for an example of the data coverage per weather station.

Only quantities with 100% data entries available were selected. See that ‘Minimum temperature’ was covered only sparsely and was therefore left out completely. After the first cleansing of the weather stations, all the remaining weather stations provided data in the first four quantities. All the weather stations therefore belong to the Group 1 which covers four weather conditions: Average wind speed, Highest wind speed, Temperature and Precipitation. The second group of weather stations covers all these conditions too and above that also Horizontal view distance, Mist, Rain, Snow, Storm and Ice. The Group 2 consist of weather stations highlighted by dark filling of the rows in Table 49. An overview of the weather conditions covered by each group and the weather stations belonging to each group is in Table 50, where 1 means that the weather condition is covered by the group and zero otherwise.

Table 45 Weather stations identification example.

STN	LON(east)	LAT(north)	ALT(m)	NAME
209	4.518	52.465	0.00	IJMOND
210	4.430	52.171	-0.20	VALKENBURG
225	4.555	52.463	4.40	IJMUIDEN

Table 46 Weather stations identification glossary.

Name	Type	Description
STN	Integer	Station number
LON	Double	Longitude (east)
LAT	Double	Latitude (north)
ALT	Double	Altitude
NAME	String	Station name

Table 47 An example of the infrastructure locations position dataset.

Name	Abbreviation	Type	Geocode	Subcode	KmLint	Kmwaarde	Latitude	Longitude	X	Y
Leeuwarden...	Lwsks	Switch	1	a	Hlg-Nscg	24.25...	53.192...	5.763...	180135.7...	578487.0...
Harinxmakanaal...	Hrm	Bridge	1	a	Hlg-Nscg	24.01...	53.191...	5.760...	179917.6...	578404.4...
Deinum	Dei	Stop	1	a	Hlg-Nscg	21.83...	53.188...	5.727...	177741.2...	578052.9...
Zevenaar	Zv	Station	611	b	Asa-Zvg	105.86...	51.923...	6.072...	202110.4...	437405.1...

Table 48 Infrastructure locations position dataset glossary

Name	Type	Description
Name	String	Location name
Abbreviation	String	
Type	String	Type of the infrastructure point
Geocode	Integer	
Subcode	String	
KmLint	String	
Kmwaarde	Float	Precision of 6 decimal points
Latitude, Longitude	Float	Degrees with precision of 12 decimal points
X, Y	Float	Precision of 6 decimal points

Table 49 Weather data stations selection: weather conditions data coverage [%]

Station number	Average wind speed	Highest wind speed	Temperature	Precipitation	Minimum temperature	Horizontal view distance	Mist	Rain	Snow	Storm	Ice
269	100	100	100	100	17	100	100	100	100	100	100
270	100	100	100	100	0	98	98	98	98	98	98
273	100	100	100	100	17	100	100	100	100	100	100
277	100	100	100	100	0	0	0	0	0	0	0

Table 50 Weather conditions considered in the groups of weather stations

	Average wind speed	Highest wind speed	Temperature	Minimum temperature	Precipitation	Horizontal view distance	Mist	Rain	Snow	Storm	Ice	Stations: original number identification by KNMI (KNMI, n.d.-b)
Group 1	1	1	1	0	1	0	0	0	0	0	0	215, 235, 240, 249, 251, 260, 267, 269, 270, 273, 275, 277, 278, 279, 280, 283, 286, 290, 310, 319, 330, 344, 348, 350, 356, 370, 375, 377, 380, 391
Group 2	1	1	1	0	1	1	1	1	1	1	1	215, 235, 240, 249, 251, 260, 269, 273, 275, 279, 280, 290, 310, 319, 344, 348, 350, 370, 375, 377, 380

Each of the infrastructure locations was to be coupled with the nearest weather station from each group. The distances between the weather stations and the infrastructure locations were calculated using Haversine formula (A.1). The formula gives a great-circle distance between two points on the Earth's surface using their longitudes and latitudes (Chopde & Nichat, 2013).

$$d = 2 \cdot r \cdot \sin^{-1}\left(\sqrt{\sin^2\left(\frac{\emptyset_2 - \emptyset_1}{2}\right) + \cos(\emptyset_1) \cdot \cos(\emptyset_2) \cdot \sin^2\left(\frac{\psi_2 - \psi_1}{2}\right)}\right) \quad (A.1)$$

Where d is the great-circle distance between the two points, r is the radius of the Earth (considered to be 6,378 km (Wikipedia, 2019)) and Ø and ψ are latitude and longitude in radians respectively. The distance was calculated between all pairs of weather stations and infrastructure locations. For each infrastructure location, only the nearest weather station from each group was stored. The distances were not stored for any further use.

Finally, the weather data were paired with corresponding realization data based on a matching timestamp (Year-Month-Day-Hour) and weather station number. The resulting two data files contained equal number of rows as the realization data, and columns as was the number of weather conditions associated with the relevant group of weather stations.

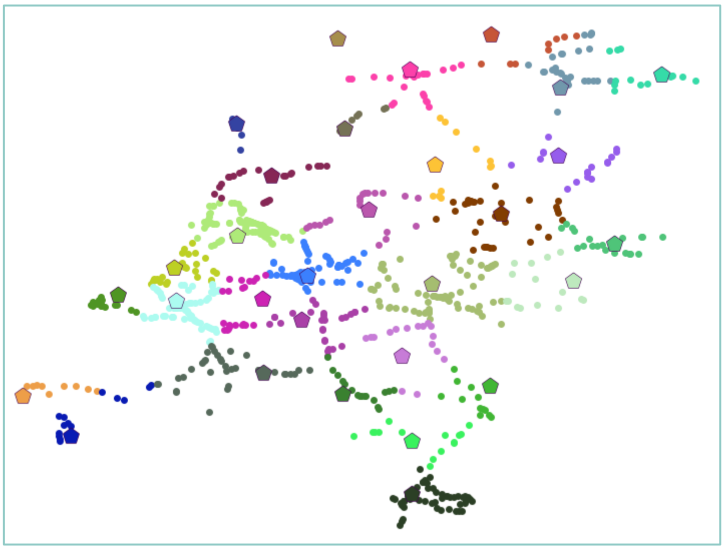


Figure 44 Pairs of weather stations (hexagon shaped) and assigned infrastructure locations.

Example illustrated on the Group 1.

A.6. Data overview and analysis

Table 51 List of train series as operated by NS that were considered in our research.

Train Series	Train Type	Train Series	Train Type	Train Series	Train Type	Train Series	Train Type	Train Series	Train Type
500	IC	3100	IC	3300	SPR	6100	SPR	8500	SPR
600	IC	3400	IC	4000	SPR	6200	SPR	8600	SPR
700	IC	3500	IC	4300	SPR	6300	SPR	8700	SPR
800	IC	3600	IC	4400	SPR	6400	SPR	8900	SPR
1400	IC	4500	IC	4600	SPR	6600	SPR	9000	SPR
1500	IC	8800	IC	4700	SPR	6700	SPR	9600	SPR
1600	IC	11400	IC	4800	SPR	6800	SPR	14000	SPR
1700	IC	11600	IC	4900	SPR	6900	SPR	14600	SPR
1800	IC	11700	IC	5000	SPR	7000	SPR	14900	SPR
2000	IC	12200	IC	5100	SPR	7300	SPR	15800	SPR
2100	IC	12600	IC	5200	SPR	7400	SPR	5300	ST
2200	IC	13500	IC	5500	SPR	7500	SPR	20300	ST
2300	IC	14500	IC	5600	SPR	7600	SPR		
2400	IC	21400	IC	5700	SPR	7700	SPR		
2600	IC	23500	IC	5800	SPR	7800	SPR		
2800	IC	24400	IC	5900	SPR	7900	SPR		
3000	IC			6000	SPR	8100	SPR		

B. Results

B.1. Regression

Table 52 Regression results: RMSE per train series (IC type) and feature set.

B	Naïve	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
L			0	1	1	1	1	1	1	1	1	1	1	1	1	1
P					0	1	2	3	4							0
W										0	1	2				0
I													3	4	5	3
Train series																
500	2.20	1.78	1.72	1.72	1.75	1.75	1.75	1.74	1.74	1.74	1.76	1.75	1.69	1.69	1.71	1.72
600	2.12	1.63	1.59	1.59	1.59	1.60	1.60	1.60	1.60	1.62	1.61	1.61	1.55	1.54	1.59	1.59
700	2.15	1.61	1.59	1.59	1.60	1.59	1.60	1.59	1.59	1.61	1.61	1.60	1.59	1.59	1.59	1.59
800	2.26	1.67	1.66	1.65	1.66	1.67	1.66	1.67	1.67	1.66	1.67	1.67	1.65	1.65	1.65	1.66
1400	2.85	2.32	2.24	2.21	2.27	2.21	2.20	2.17	2.16	2.15	2.14	2.14	2.18	2.18	2.18	2.16
1500	2.00	1.73	1.71	1.71	1.75	1.75	1.75	1.74	1.75	1.77	1.77	1.77	1.70	1.69	1.70	1.78
1600	2.18	1.74	1.72	1.73	1.75	1.75	1.75	1.76	1.75	1.74	1.73	1.74	1.71	1.71	1.72	1.73
1700	2.24	1.82	1.81	1.80	1.82	1.82	1.82	1.82	1.82	1.83	1.84	1.84	1.78	1.79	1.80	1.81
1800	2.06	1.56	1.55	1.54	1.55	1.55	1.55	1.55	1.54	1.55	1.56	1.56	1.54	1.54	1.54	1.55
2000	1.98	1.47	1.45	1.45	1.46	1.46	1.46	1.46	1.45	1.47	1.47	1.46	1.37	1.39	1.44	1.41
2100	1.91	1.40	1.38	1.38	1.38	1.39	1.40	1.39	1.38	1.42	1.41	1.41	1.36	1.36	1.38	1.37
2200	2.25	1.70	1.67	1.67	1.68	1.69	1.69	1.69	1.69	1.67	1.68	1.68	1.66	1.66	1.67	1.67
2300	2.15	1.91	1.89	1.89	1.89	1.89	1.89	1.89	1.89	1.91	1.94	1.94	1.84	1.85	1.88	1.86
2400	2.20	1.72	1.71	1.70	1.71	1.72	1.72	1.72	1.72	1.72	1.72	1.72	1.69	1.69	1.71	1.70
2600	1.61	1.59	1.58	1.58	1.61	1.62	1.61	1.61	1.61	1.66	1.65	1.66	1.56	1.57	1.58	1.64
2800	1.74	1.46	1.43	1.43	1.47	1.46	1.45	1.45	1.46	1.46	1.46	1.46	1.41	1.41	1.43	1.44
3000	2.11	1.53	1.44	1.44	1.44	1.45	1.45	1.46	1.46	1.44	1.44	1.44	1.42	1.42	1.43	1.43
3100	2.27	1.66	1.66	1.66	1.67	1.67	1.68	1.69	1.69	1.68	1.68	1.68	1.65	1.66	1.67	1.66
3400	2.05	1.55	1.51	1.51	1.59	1.43	1.44	1.41	1.42	1.43	1.41	1.41	1.42	1.44	1.47	1.41
3500	1.98	1.29	1.25	1.24	1.25	1.25	1.24	1.25	1.26	1.24	1.25	1.25	1.24	1.25	1.26	1.24
3600	2.21	1.72	1.71	1.71	1.71	1.72	1.71	1.72	1.72	1.70	1.70	1.70	1.68	1.68	1.71	1.69
4500	1.74	1.26	1.26	1.25	1.26	1.26	1.27	1.27	1.27	1.25	1.24	1.24	1.25	1.25	1.26	1.25
8800	1.75	1.29	1.27	1.27	1.28	1.29	1.28	1.28	1.28	1.27	1.27	1.27	1.26	1.26	1.26	1.27
11400	1.42	1.23	1.18	1.18	1.25	1.14	1.19	1.19	1.20	1.24	1.27	1.26	1.15	1.15	1.15	1.22
11600	1.83	1.31	1.30	1.29	1.30	1.30	1.30	1.32	1.33	1.32	1.33	1.32	1.30	1.30	1.30	1.32
11700	2.04	1.74	1.71	1.71	1.72	1.70	1.70	1.69	1.71	1.73	1.73	1.74	1.67	1.68	1.69	1.70
12200	1.36	0.94	1.03	1.03	0.95	1.09	1.06	1.09	1.04	1.16	1.16	1.11	1.02	1.00	1.00	1.09
12600	2.55	2.30	2.29	2.28	2.28	2.35	2.25	2.44	2.33	2.22	2.31	2.31	2.27	2.27	2.27	2.29
13500	1.76	1.36	1.35	1.34	1.36	1.37	1.38	1.37	1.38	1.41	1.42	1.43	1.34	1.34	1.34	1.41
14500	2.26	1.62	1.54	1.53	1.62	1.61	1.61	1.61	1.67	1.66	1.70	1.70	1.52	1.52	1.52	1.66
21400	3.08	2.55	2.07	2.04	2.37	2.10	2.09	1.99	1.89	2.00	2.03	2.02	1.93	1.93	1.93	2.06
23500	0.97	0.50	0.49	0.49	0.48	0.51	0.50	0.51	0.50	0.67	0.63	0.62	0.49	0.49	0.49	0.61
24400	1.50	1.43	1.44	1.44	1.48	1.52	1.46	1.51	1.47	1.53	1.55	1.55	1.44	1.44	1.44	1.56

Table 53 Regression results: RMSE per train series (SPR type) and feature set.

B	Naïve	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
L			0	1	1	1	1	1	1	1	1	1	1	1	1	1
P					0	1	2	3	4							0
W										0	1	2				0
I													3	4	5	3
Train series																
3300	1.92	1.27	1.27	1.27	1.27	1.29	1.28	1.29	1.28	1.30	1.29	1.29	1.27	1.27	1.27	1.29
4000	1.96	1.47	1.47	1.47	1.45	1.44	1.44	1.44	1.44	1.44	1.44	1.44	1.46	1.45	1.47	1.43
4300	1.80	1.30	1.27	1.27	1.28	1.25	1.26	1.29	1.28	1.26	1.25	1.25	1.27	1.27	1.27	1.26
4400	2.16	1.72	1.70	1.70	1.72	1.73	1.72	1.73	1.72	1.71	1.71	1.71	1.70	1.70	1.69	1.72
4600	1.89	1.46	1.44	1.44	1.44	1.44	1.44	1.44	1.44	1.42	1.42	1.42	1.43	1.44	1.43	1.42
4700	1.82	1.37	1.37	1.37	1.38	1.38	1.38	1.39	1.38	1.41	1.42	1.42	1.36	1.36	1.36	1.40
4800	2.02	1.57	1.57	1.56	1.57	1.58	1.58	1.59	1.59	1.57	1.58	1.58	1.57	1.56	1.56	1.58
4900	1.69	1.08	1.06	1.06	1.09	1.09	1.10	1.10	1.10	1.12	1.12	1.12	1.05	1.05	1.06	1.10
5000	1.80	1.37	1.35	1.35	1.37	1.37	1.37	1.37	1.37	1.39	1.39	1.39	1.34	1.34	1.34	1.38
5100	1.76	1.28	1.26	1.26	1.26	1.26	1.26	1.26	1.26	1.26	1.26	1.26	1.25	1.25	1.25	1.25
5200	1.69	1.50	1.49	1.49	1.50	1.49	1.49	1.48	1.49	1.49	1.50	1.50	1.48	1.48	1.48	1.48
5500	1.60	1.13	1.12	1.12	1.13	1.15	1.15	1.14	1.14	1.13	1.13	1.13	1.11	1.11	1.12	1.13
5600	2.11	1.63	1.61	1.62	1.61	1.61	1.63	1.62	1.62	1.62	1.63	1.64	1.62	1.62	1.62	1.61
5700	1.66	1.29	1.26	1.25	1.26	1.27	1.27	1.26	1.23	1.24	1.23	1.23	1.26	1.26	1.26	1.21
5800	1.93	1.34	1.33	1.33	1.35	1.35	1.35	1.35	1.35	1.34	1.35	1.34	1.31	1.31	1.33	1.30
5900	1.80	1.42	1.40	1.40	1.43	1.43	1.44	1.42	1.42	1.42	1.42	1.42	1.42	1.41	1.41	1.45
6000	1.71	1.40	1.39	1.39	1.41	1.40	1.41	1.41	1.40	1.42	1.41	1.42	1.38	1.39	1.39	1.41
6100	1.35	1.09	1.08	1.08	1.09	1.09	1.09	1.10	1.10	1.08	1.08	1.08	1.07	1.07	1.07	1.09
6200	1.81	1.78	1.77	1.76	1.68	1.66	1.58	1.68	1.67	1.54	1.53	1.52	1.73	1.75	1.76	1.51
6300	1.76	1.14	1.13	1.13	1.13	1.13	1.14	1.13	1.13	1.12	1.12	1.12	1.13	1.13	1.13	1.12
6400	1.77	1.52	1.51	1.52	1.54	1.57	1.56	1.58	1.61	1.56	1.55	1.56	1.50	1.51	1.50	1.56
6600	2.08	1.61	1.60	1.60	1.61	1.62	1.62	1.62	1.62	1.62	1.64	1.64	1.61	1.61	1.61	1.62
6700	1.05	1.32	1.33	1.33	1.30	1.33	1.34	1.31	1.40	1.35	1.41	1.39	1.33	1.33	1.33	1.34
6800	1.27	1.00	0.98	0.99	1.00	0.97	0.98	0.97	0.97	0.97	0.95	0.95	0.95	0.96	0.96	0.96
6900	1.84	1.59	1.59	1.59	1.60	1.61	1.58	1.59	1.59	1.60	1.57	1.57	1.57	1.57	1.57	1.59
7000	1.97	1.37	1.35	1.35	1.36	1.37	1.36	1.35	1.35	1.36	1.37	1.37	1.34	1.34	1.35	1.35
7300	2.15	1.95	1.93	1.93	1.93	1.93	1.93	1.92	1.91	1.94	1.93	1.94	1.90	1.90	1.91	1.92
7400	1.98	1.38	1.36	1.36	1.37	1.39	1.39	1.38	1.39	1.40	1.41	1.41	1.36	1.35	1.36	1.38
7500	1.47	1.35	1.35	1.35	1.36	1.38	1.38	1.38	1.38	1.35	1.36	1.35	1.35	1.35	1.35	1.35
7600	1.87	1.45	1.43	1.43	1.46	1.41	1.42	1.43	1.42	1.48	1.47	1.46	1.40	1.40	1.41	1.47
7700	1.42	1.15	1.12	1.12	1.16	1.19	1.16	1.18	1.20	1.26	1.28	1.25	1.13	1.09	1.14	1.24
7800	2.00	1.47	1.44	1.44	1.45	1.47	1.46	1.46	1.47	1.44	1.44	1.44	1.44	1.44	1.44	1.44
7900	2.01	1.43	1.42	1.42	1.43	1.46	1.44	1.45	1.45	1.40	1.41	1.41	1.41	1.41	1.41	1.38
8100	1.42	1.03	1.02	1.01	1.02	1.05	1.11	1.04	1.03	1.06	1.06	1.07	1.02	1.01	1.01	1.06
8500	1.54	1.00	1.00	1.00	0.97	1.14	1.07	1.05	1.02	0.95	0.93	0.93	0.99	0.99	0.99	0.94
8600	1.71	1.24	1.20	1.21	1.21	1.22	1.21	1.25	1.22	1.18	1.17	1.17	1.22	1.22	1.21	1.19
8700	1.43	0.85	0.84	0.84	0.84	0.81	0.81	0.81	0.82	0.83	0.82	0.82	0.82	0.83	0.83	0.82
8900	1.69	1.41	1.40	1.40	1.40	1.41	1.40	1.40	1.41	1.39	1.38	1.39	1.39	1.39	1.39	1.40
9000	1.68	1.33	1.32	1.33	1.32	1.33	1.33	1.32	1.32	1.33	1.33	1.33	1.32	1.32	1.32	1.33
9600	2.00	1.56	1.55	1.54	1.56	1.56	1.55	1.56	1.56	1.56	1.57	1.57	1.55	1.55	1.55	1.56
14000	2.18	1.79	1.80	1.81	1.72	1.65	1.64	1.69	1.66	1.78	1.76	1.76	1.76	1.76	1.76	1.73
14600	2.03	1.47	1.45	1.44	1.45	1.45	1.45	1.45	1.44	1.45	1.44	1.44	1.45	1.44	1.45	1.44
14900	1.72	1.25	1.24	1.24	1.25	1.26	1.30	1.25	1.27	1.28	1.27	1.26	1.22	1.22	1.22	1.28
15800	2.11	1.70	1.67	1.67	1.69	1.69	1.69	1.70	1.70	1.68	1.69	1.69	1.66	1.67	1.67	1.68
5300	1.29	1.05	1.04	1.04	1.03	1.03	1.03	1.03	1.03	1.00	1.00	1.00	1.04	1.04	1.04	1.00
20300	0.93	0.33	0.32	0.32	0.32	0.32	0.33	0.33	0.33	0.31	0.31	0.32	0.32	0.32	0.32	0.31

B.2. Discretized prediction

Table 54 F1 scores of the discretized prediction in categories of delay length.

Feature sets					Delay [min]															
B	L	P	W	I	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Naïve forecast					0.59	0.37	0.24	0.21	0.20	0.18	0.18	0.18	0.16	0.17	0.19	0.17	0.17	0.17	0.20	0.68
0					0.74	0.54	0.34	0.27	0.23	0.20	0.19	0.16	0.14	0.14	0.15	0.15	0.15	0.14	0.15	0.68
0	0				0.75	0.54	0.35	0.28	0.25	0.22	0.21	0.18	0.15	0.14	0.17	0.16	0.16	0.15	0.16	0.68
0	1				0.75	0.54	0.35	0.28	0.25	0.22	0.21	0.18	0.16	0.15	0.17	0.16	0.16	0.16	0.17	0.68
0	1	0			0.74	0.54	0.34	0.27	0.24	0.21	0.21	0.18	0.16	0.15	0.16	0.16	0.15	0.15	0.15	0.67
0	1	1			0.74	0.54	0.34	0.27	0.24	0.21	0.20	0.18	0.15	0.14	0.16	0.16	0.16	0.14	0.14	0.67
0	1	2			0.74	0.54	0.34	0.27	0.24	0.21	0.20	0.18	0.15	0.14	0.16	0.16	0.15	0.14	0.15	0.67
0	1	3			0.74	0.54	0.34	0.27	0.24	0.21	0.20	0.18	0.15	0.15	0.15	0.17	0.16	0.15	0.15	0.67
0	1	4			0.74	0.54	0.34	0.27	0.24	0.21	0.20	0.18	0.15	0.14	0.14	0.16	0.16	0.16	0.15	0.67
0	1		0		0.75	0.53	0.34	0.27	0.24	0.21	0.20	0.18	0.16	0.15	0.14	0.15	0.15	0.16	0.15	0.67
0	1		1		0.74	0.53	0.33	0.27	0.24	0.20	0.20	0.17	0.15	0.15	0.15	0.16	0.15	0.16	0.15	0.66
0	1		2		0.74	0.53	0.33	0.27	0.24	0.20	0.20	0.18	0.15	0.15	0.14	0.15	0.15	0.16	0.16	0.66
0	1			3	0.75	0.54	0.35	0.28	0.24	0.22	0.21	0.19	0.16	0.15	0.18	0.17	0.16	0.15	0.15	0.68
0	1			4	0.74	0.54	0.34	0.28	0.24	0.22	0.21	0.19	0.15	0.14	0.17	0.17	0.16	0.15	0.17	0.68
0	1			5	0.75	0.54	0.35	0.28	0.25	0.22	0.22	0.19	0.16	0.15	0.17	0.17	0.16	0.15	0.14	0.68
0	1	0	0	3	0.75	0.54	0.34	0.27	0.24	0.21	0.20	0.18	0.15	0.16	0.13	0.16	0.16	0.15	0.15	0.66
all combined					0.75	0.54	0.34	0.28	0.24	0.22	0.21	0.19	0.16	0.15	0.18	0.17	0.15	0.15	0.16	0.68
Comparison only among feature sets including 'Interactions'																				
0	1			0	0.71	0.55	0.34	0.26	0.23	0.19	0.20	0.17	0.14	0.14	0.14	0.15	0.17	0.16	0.14	0.62
0	1			1	0.71	0.55	0.34	0.26	0.23	0.19	0.19	0.17	0.14	0.14	0.14	0.15	0.16	0.15	0.15	0.81
0	1			2	0.70	0.55	0.34	0.26	0.22	0.19	0.19	0.16	0.14	0.13	0.14	0.15	0.17	0.16	0.14	0.62
0	1			3	0.71	0.55	0.34	0.26	0.22	0.19	0.20	0.17	0.14	0.14	0.15	0.16	0.16	0.15	0.15	0.81
0	1			4	0.71	0.55	0.34	0.26	0.22	0.19	0.20	0.17	0.14	0.13	0.14	0.15	0.16	0.15	0.15	0.81
0	1			5	0.70	0.54	0.34	0.26	0.22	0.19	0.19	0.16	0.14	0.12	0.14	0.15	0.17	0.15	0.18	0.62

B.3. Classification tasks

Table 55 Delay existence classification task: Complete results.

Feature sets					F1		Precision		Recall		Accuracy
B	L	P	W	I	0	1	0	1	0	1	
Naïve forecast					0.942	0.586	0.978	0.462	0.907	0.798	0.898
0					0.948	0.655	0.977	0.548	0.921	0.814	0.909
0	0				0.949	0.659	0.978	0.551	0.922	0.819	0.911
0	1				0.949	0.660	0.977	0.553	0.922	0.819	0.911
0	1	0			0.948	0.657	0.976	0.552	0.922	0.812	0.910
0	1	1			0.948	0.654	0.977	0.545	0.921	0.816	0.910
0	1	2			0.948	0.653	0.977	0.545	0.921	0.815	0.910
0	1	3			0.948	0.654	0.977	0.546	0.921	0.815	0.910
0	1	4			0.948	0.653	0.977	0.545	0.921	0.815	0.910
0	1		0		0.947	0.656	0.973	0.559	0.923	0.795	0.909
0	1		1		0.947	0.655	0.973	0.558	0.923	0.793	0.908
0	1		2		0.947	0.655	0.973	0.558	0.923	0.793	0.908
0	1			3	0.950	0.665	0.977	0.559	0.923	0.820	0.912
0	1			4	0.949	0.665	0.977	0.559	0.923	0.820	0.912
0	1			5	0.949	0.659	0.978	0.550	0.922	0.822	0.911
0	1			0	0.944	0.655	0.974	0.552	0.916	0.806	0.904
0	1			1	0.944	0.655	0.974	0.551	0.916	0.808	0.904
0	1			2	0.944	0.647	0.975	0.537	0.914	0.813	0.903
0	1			3	0.944	0.656	0.975	0.549	0.916	0.813	0.904
0	1			4	0.944	0.656	0.975	0.549	0.916	0.813	0.904
0	1			5	0.944	0.648	0.976	0.537	0.914	0.815	0.903
0	1	0	0	3	0.948	0.661	0.974	0.564	0.924	0.799	0.910
All combined					0.949	0.665	0.977	0.560	0.923	0.817	0.912

Table 56 Delay change classification task: Complete results.

Feature sets					F1			Precision			Recall			Accuracy
B	L	P	W	I	-	=	+	-	=	+	-	=	+	
0					0.422	0.928	0.217	0.309	0.980	0.135	0.667	0.881	0.551	0.868
0	0				0.445	0.929	0.225	0.328	0.980	0.141	0.688	0.883	0.567	0.870
0	1				0.453	0.929	0.229	0.337	0.980	0.143	0.691	0.884	0.565	0.871
0	1	0			0.446	0.928	0.226	0.331	0.979	0.143	0.684	0.883	0.537	0.869
0	1	1			0.427	0.928	0.215	0.310	0.981	0.133	0.687	0.881	0.554	0.869
0	1	2			0.428	0.928	0.216	0.311	0.981	0.134	0.688	0.881	0.558	0.869
0	1	3			0.428	0.928	0.215	0.311	0.981	0.133	0.687	0.881	0.553	0.869
0	1	4			0.426	0.928	0.216	0.309	0.981	0.134	0.686	0.881	0.554	0.869
0	1		0		0.457	0.927	0.241	0.345	0.975	0.159	0.676	0.885	0.501	0.868
0	1		1		0.451	0.927	0.239	0.339	0.974	0.157	0.674	0.884	0.497	0.867
0	1		2		0.452	0.927	0.240	0.340	0.974	0.158	0.675	0.884	0.498	0.868
0	1			3	0.451	0.930	0.245	0.332	0.980	0.155	0.701	0.884	0.579	0.872
0	1			4	0.454	0.930	0.245	0.336	0.980	0.156	0.698	0.884	0.575	0.872
0	1			5	0.446	0.929	0.225	0.328	0.981	0.139	0.699	0.883	0.579	0.871
0	1			0	0.427	0.922	0.283	0.309	0.976	0.188	0.695	0.874	0.568	0.859
0	1			1	0.426	0.922	0.281	0.307	0.976	0.186	0.692	0.874	0.568	0.859
0	1			2	0.405	0.922	0.253	0.286	0.980	0.161	0.696	0.870	0.587	0.859
0	1			3	0.417	0.922	0.276	0.296	0.979	0.180	0.702	0.872	0.593	0.860
0	1			4	0.424	0.922	0.275	0.304	0.978	0.180	0.697	0.873	0.586	0.860
0	1			5	0.409	0.922	0.253	0.290	0.980	0.161	0.696	0.871	0.591	0.859
0	1	0	0	3	0.455	0.928	0.253	0.342	0.975	0.167	0.681	0.885	0.521	0.868
All combined					0.460	0.929	0.250	0.343	0.979	0.160	0.698	0.884	0.574	0.871

Table 57 Delay jump classification task: Complete results.

Feature sets					F1		Precision		Recall		Accuracy
B	L	P	W	I	0	1	0	1	0	1	
0					0.991	0.072	0.999	0.040	0.983	0.390	0.982
0	0				0.991	0.083	0.999	0.046	0.984	0.442	0.983
0	1				0.991	0.090	0.999	0.050	0.984	0.449	0.983
0	1	0			0.991	0.084	0.999	0.047	0.984	0.405	0.983
0	1	1			0.991	0.072	0.999	0.039	0.984	0.412	0.983
0	1	2			0.991	0.071	0.999	0.039	0.984	0.411	0.983
0	1	3			0.991	0.073	0.999	0.040	0.984	0.418	0.983
0	1	4			0.991	0.071	0.999	0.039	0.984	0.409	0.983
0	1		0		0.991	0.090	0.999	0.051	0.984	0.379	0.982
0	1		1		0.991	0.089	0.999	0.050	0.984	0.390	0.983
0	1		2		0.991	0.086	0.999	0.049	0.984	0.372	0.982
0	1			3	0.991	0.090	0.999	0.050	0.984	0.470	0.983
0	1			4	0.991	0.090	0.999	0.050	0.984	0.457	0.983
0	1			5	0.991	0.087	0.999	0.048	0.984	0.473	0.983
0	1			0	0.990	0.065	0.999	0.036	0.981	0.356	0.980
0	1			1	0.990	0.063	0.999	0.035	0.981	0.361	0.980
0	1			2	0.990	0.051	0.999	0.027	0.981	0.352	0.980
0	1			3	0.990	0.057	0.999	0.031	0.981	0.393	0.980
0	1			4	0.990	0.052	0.999	0.028	0.981	0.374	0.980
0	1			5	0.990	0.055	0.999	0.030	0.981	0.373	0.980
0	1	0	0	3	0.991	0.088	0.999	0.050	0.984	0.393	0.982
All combined					0.991	0.091	0.999	0.050	0.984	0.476	0.983

B.4. Features importance

Table 58 Features importance: feature sets B0, B0L0-1.

	B	L	P	W	I	DOW	Location_nr	Direction	Delay	Hour	Minutes	Delay_1before	Delay_2before	L0 / L1: V	L0 / L1: K_V	L0 / L1: D
Gain	0					0.05	0.11	0.17	0.97	0.07	0.08	0.31	0.10			
	0	0				0.05	0.08	0.14	0.99	0.08	0.06	0.40	0.12	0.14	0.12	0.13
	0	1				0.05	0.07	0.14	0.99	0.08	0.06	0.40	0.12	0.14	0.11	0.12
Weig	0					0.46	0.68	0.23	0.83	0.89	0.91	0.67	0.65			
	0	0				0.45	0.53	0.19	0.92	0.89	0.83	0.66	0.63	0.15	0.20	0.34
	0	1				0.45	0.45	0.17	0.93	0.89	0.79	0.66	0.62	0.21	0.26	0.43
Cover	0					0.30	0.78	0.48	0.97	0.51	0.61	0.62	0.55			
	0	0				0.31	0.74	0.42	0.96	0.51	0.55	0.60	0.55	0.61	0.59	0.64
	0	1				0.31	0.71	0.42	0.96	0.51	0.53	0.60	0.55	0.59	0.61	0.70

Table 59 Features importance: feature sets B0L1P0-4.

	B	L	P	W	I	DOW	Location_nr	Direction	Delay	Hour	Minutes	Delay_1before	Delay_2before	L0 / L1: V	L0 / L1: K_V	L0 / L1: D	seats_ratio	peak_dep	Pax_Total_1-4	Pax_Board_1-4	Pax_Alight_1-4
Gain	0	1	0			0.05	0.07	0.13	0.99	0.08	0.06	0.33	0.11	0.13	0.10	0.12	0.05	0.05			
	0	1	1			0.04	0.06	0.11	1.00	0.06	0.05	0.39	0.12	0.10	0.08	0.09	0.05	0.04	0.05	0.05	0.05
	0	1	2			0.04	0.06	0.11	1.00	0.06	0.05	0.39	0.12	0.10	0.08	0.09	0.04	0.04	0.05	0.04	0.04
	0	1	3			0.04	0.06	0.11	1.00	0.06	0.05	0.39	0.12	0.10	0.08	0.09	0.05	0.04	0.05	0.04	0.05
	0	1	4			0.04	0.06	0.11	1.00	0.06	0.05	0.39	0.12	0.10	0.08	0.10	0.05	0.04	0.05	0.05	0.05
Weight	0	1	0			0.47	0.43	0.16	0.80	0.91	0.79	0.56	0.54	0.19	0.23	0.38	0.55	0.19			
	0	1	1			0.33	0.38	0.15	0.64	0.69	0.62	0.38	0.34	0.19	0.23	0.37	0.30	0.10	0.99	0.87	0.92
	0	1	2			0.33	0.38	0.16	0.65	0.70	0.63	0.39	0.35	0.19	0.23	0.38	0.30	0.10	0.98	0.89	0.93
	0	1	3			0.33	0.38	0.16	0.64	0.70	0.62	0.38	0.34	0.19	0.23	0.37	0.30	0.10	0.97	0.87	0.95
	0	1	4			0.33	0.38	0.16	0.65	0.70	0.62	0.38	0.34	0.19	0.23	0.37	0.30	0.10	0.98	0.87	0.93
Cover	0	1	0			0.23	0.52	0.39	0.97	0.40	0.36	0.50	0.39	0.53	0.54	0.59	0.58	0.22			
	0	1	1			0.19	0.42	0.33	0.98	0.35	0.27	0.48	0.34	0.44	0.45	0.47	0.51	0.20	0.32	0.33	0.33
	0	1	2			0.19	0.43	0.33	0.99	0.35	0.27	0.49	0.34	0.44	0.44	0.46	0.52	0.20	0.32	0.33	0.34
	0	1	3			0.18	0.42	0.33	0.99	0.35	0.27	0.49	0.34	0.43	0.45	0.46	0.51	0.19	0.32	0.33	0.35
	0	1	4			0.18	0.42	0.33	0.98	0.35	0.27	0.49	0.34	0.44	0.45	0.47	0.51	0.19	0.32	0.34	0.34

Table 60 Features importance: feature sets B0L1W0-2.

	B	L	P	W	I	DOW	Location_nr	Direction	Delay	Hour	Minutes	Delay_1before	Delay_2before	L0 / L1: V	L0 / L1: K_V	L0 / L1: D	Avgwind	Highwind	Temp	Percp	View	Mist	Rain	Snow	Storm	Ice	BadWeather
Gain	0	1		0		0.05	0.06	0.11	1.00	0.07	0.06	0.31	0.09	0.10	0.08	0.10	0.05	0.05	0.06	0.06							
	0	1		1		0.05	0.06	0.11	1.00	0.07	0.05	0.38	0.11	0.10	0.09	0.10	0.05	0.05	0.06	0.06	0.05	0.03	0.05	0.04	0.03	0.03	
	0	1		2		0.05	0.06	0.11	1.00	0.07	0.05	0.35	0.11	0.10	0.08	0.10	0.05	0.05	0.06	0.06	0.05						0.05
Weig	0	1		0		0.30	0.29	0.15	0.51	0.63	0.61	0.33	0.32	0.14	0.17	0.29	0.38	0.45	0.99	0.29							
	0	1		1		0.31	0.34	0.18	0.58	0.66	0.70	0.37	0.35	0.17	0.20	0.34	0.39	0.45	0.99	0.27	0.64	0.01	0.06	0.01	0.01	0.01	
	0	1		2		0.31	0.34	0.17	0.57	0.66	0.70	0.37	0.35	0.17	0.21	0.34	0.39	0.45	0.99	0.28	0.64						0.09
Cover	0	1		0		0.20	0.48	0.31	0.99	0.38	0.31	0.47	0.35	0.46	0.48	0.53	0.3	0.34	0.45	0.44							
	0	1		1		0.15	0.35	0.23	0.76	0.29	0.23	0.37	0.27	0.33	0.35	0.38	0.22	0.26	0.32	0.31	0.28	0.36	0.18	0.37	0.6	0.28	
	0	1		2		0.19	0.47	0.30	0.99	0.38	0.30	0.49	0.35	0.45	0.46	0.50	0.29	0.34	0.43	0.42	0.37						0.28

Table 61 Features importance: feature sets B0L1I3-5.

	B	L	P	W	I	DOW	Location_nr	Direction	Delay	Hour	Minutes	Delay_1before	Delay_2before	L0 / L1: V	L0 / L1: K_V	L0 / L1: D	IC	SPR	LM
Gain	0	1			3	0.04	0.07	0.13	0.99	0.07	0.05	0.36	0.11	0.13	0.10	0.11	0.04	0.03	0.00
	0	1			4	0.04	0.06	0.13	0.99	0.07	0.05	0.36	0.11	0.13	0.10	0.11	0.05	0.04	0.00
	0	1			5	0.04	0.07	0.13	0.99	0.07	0.05	0.36	0.11	0.13	0.10	0.11	0.03	0.03	0.00
Weigh	0	1			3	0.39	0.41	0.15	0.93	0.82	0.70	0.57	0.52	0.20	0.24	0.40	0.34	0.27	0.00
	0	1			4	0.40	0.43	0.16	0.93	0.83	0.72	0.59	0.54	0.20	0.25	0.41	0.16	0.17	0.00
	0	1			5	0.38	0.42	0.16	0.96	0.82	0.71	0.60	0.54	0.21	0.25	0.42	0.07	0.05	0.00
Cover	0	1			3	0.30	0.70	0.38	0.90	0.49	0.49	0.55	0.52	0.54	0.56	0.67	0.50	0.43	0.13
	0	1			4	0.33	0.72	0.39	0.92	0.52	0.53	0.58	0.55	0.54	0.57	0.68	0.43	0.37	0.13
	0	1			5	0.30	0.70	0.38	0.90	0.49	0.49	0.55	0.52	0.54	0.56	0.67	0.50	0.43	0.13