Advances in Science and Technology
Research Journal

Advances in Science and Technology Research Journal, 2025, 19(11), 382–399
https://doi.org/10.12913/22998624/209290
ISSN 2299-8624, License CC-BY 4.0

Received: 2025.05.30
Accepted: 2025.09.15
Published: 2025.10.01

Optimizing traffic volume prediction: Linear regression vs. random forest

Paweł Dymora[1*], Mirosław Mazurek[1], Maksymilian Jucha[1]

[1] Faculty of Electrical and Computer Engineering, Rzeszów University of Technology, Al. Powstańców Warszawy 12, 35-959 Rzeszów, Poland

* Corresponding author's e-mail: pawel.dymora@prz.edu.pl

ABSTRACT

In this work, two series of regression models were constructed and tested – one comprising of models based on the random forest algorithm, a machine learning method, and the other based on linear regression. The models were fitted to the data on traffic flow within chosen intersections in the city of Rzeszów and optimized by manipulating the explanatory variables and input parameters. Construction process and optimization efforts have been extensively documented in this article. The performance of both types of models was evaluated in a series of tests, including fitness to the empirical data, residual distribution, prediction of new data, and model training time. Both kinds of models passed the tests favourably, while pointing out some of the advantages and disadvantages of the regression methods used. The results are illustrated on various charts, and the most interesting parts of the program code used are presented.

Keywords: machine learning, linear regression, random forest, traffic flow.

INTRODUCTION

Prediction, understood as the process of forecasting future values based on historical data, is a key analytical tool widely applied in various domains, particularly in traffic flow management. Short-term prediction is especially important, as it enables real-time responses to dynamic conditions, such as adjusting traffic signals, optimizing vehicle routing, or managing traffic during emergencies. In recent years, regression models have attracted significant attention as effective tools for short-term traffic prediction.

Accurate and timely prediction of traffic flow is a cornerstone of modern intelligent transportation systems (ITS), particularly in the context of smart cities and adaptive traffic management. With the growth of urban populations and increasing vehicle density, traffic congestion has become a critical societal and economic issue. Consequently, the development of reliable models for short-term traffic prediction has become an active area of research. These models support real-time decisions, including dynamic traffic light control, route optimization, congestion mitigation, and emergency management strategies.

Traditional approaches to traffic prediction, such as time series analysis and linear regression, have proven effective under certain conditions, particularly when traffic patterns exhibit strong seasonality or periodicity. However, the complexity and variability of real-world traffic, influenced by factors such as weather, accidents, infrastructure, and human behavior, demand more flexible, data-driven methods. In response, machine learning techniques have gained significant traction in traffic prediction research due to their ability to model non-linear relationships and handle high-dimensional datasets.

The prediction of traffic flow has evolved significantly over the decades, starting from classical statistical tools to advanced machine learning approaches. One of the foundational contributions to statistical analysis was made by Shapiro and Wilk

[1], who proposed a test for assessing the normality of a dataset. While their method is not directly used for traffic forecasting, it plays a crucial role in validating assumptions before applying more complex models. Early work on traffic prediction emphasized probabilistic models, notably Bayesian methods. Zheng et al. [2] introduced a Bayesian combined neural network approach that improved short-term freeway traffic predictions by integrating multiple neural networks, achieving more robust generalization. Similarly, Sun et al. [3] proposed a Bayesian network model capable of handling uncertainty and variable dependencies in traffic flow. These models marked a shift toward data-driven, probabilistic approaches that accounted for real-world variability.

The development of real-time prediction models was advanced by Min and Wynter [4], who incorporated spatio-temporal correlations. Their approach reflected the growing complexity of urban traffic systems and the need to model interactions across time and space. Another significant methodology involves dynamic estimation using Kalman filtering. Wang et al. [5] employed the extended Kalman filter (EKF) to estimate real-time freeway states, showing its utility in modeling latent traffic variables like density and speed. The relevance of this topic is also reflected in numerous other studies. For instance, Lv and Duan [6], proposed the use of deep learning models for short-term traffic prediction, demonstrating their superiority over traditional methods when dealing with large and complex datasets. Similarly, Vlahogianni and Karlaftis [7] emphasized the effectiveness of hybrid approaches that combine statistical methods with machine learning techniques to enhance forecast accuracy under variable traffic conditions. Yang and Pan [8] showed that integrating data from diverse sources – such as road sensors, GPS, and weather data, significantly improves prediction performance. In this context, models based on long short-term memory (LSTM) neural networks have been developed to effectively handle temporal dependencies and fluctuations in traffic volume.

Regression models, particularly those enhanced by machine learning, such as random forest regression, continue to play a central role in traffic flow modeling due to their robustness, simplicity, and interpretability. For instance, studies by Dymora et al. [9, 10] evaluated the effectiveness of classical and machine learning regression models in forecasting short-term traffic volumes within smart city environments. Their results highlight the trade-offs between computational cost and accuracy, underscoring the relevance of optimization in model construction and parameter tuning. Moreover, ensemble learning methods like random forest have shown superior generalization in traffic modeling tasks. As demonstrated by Liaw and Wiener [11], random forest s effectively handle overfitting and provide internal error estimates, making them attractive for real-world deployment. Despite their advantages, systematic comparisons between machine learning regression models and classical linear models are still relatively scarce in the context of urban traffic data with high temporal granularity. In more recent years, the use of recurrent neural networks has gained traction.

In parallel, ensemble learning techniques such as random forests (RF) have been explored for traffic prediction under complex conditions. Xu et al. [12] applied RF to forecast traffic during severe weather, highlighting the model's robustness and interpretability. More recently, Sun et al. [13] applied RF for congestion prediction, demonstrating its superior performance compared to linear regression models, particularly in ranking the importance of input features.

This study contributes to the ongoing discourse by providing a comparative analysis of two types of regression models: one based on the random forest algorithm and another on classical linear regression. The models are trained and tested using empirical traffic data collected from intersections in the city of Rzeszów, Poland. Unlike previous studies that focus solely on prediction accuracy, this work evaluates a comprehensive set of statistical indicators, including R-squared, residual distribution metrics (mean, standard deviation, kurtosis, normality), model stability across resampling, and computational performance [14, 15]. The scientific novelty of this work lies in its multi-faceted methodological framework and practical orientation. Specifically, it introduces:

- The use of Fourier-transformed time variables to encode weekly cyclic patterns in traffic data;
- A systematic hyperparameter tuning process for random forest using grid search;
- A detailed evaluation of model performance trade-offs between predictive accuracy and computational cost.

Furthermore, the study highlights that even lightweight, interpretable models such as linear regression – when properly optimized – can perform competitively against more complex alternatives. This insight is especially relevant for real-world applications where computational resources may be constrained. In addition, this research addresses a frequently overlooked aspect in the literature: the interpretability-performance trade-off in traffic forecasting models, offering a reproducible and scalable methodology that balances simplicity with predictive strength.

The primary objective of this study is to develop and evaluate a statistically robust regression model for predicting weekly traffic volumes, utilizing both classical and machine learning approaches. By comparing a random forest -based model with a linear regression counterpart, the research addresses the question of whether increased model complexity (in terms of algorithmic sophistication and computational cost) yields proportionally improved predictive power in real-world, time-indexed traffic datasets. The main objective of this study is to construct a statistically significant regression model describing weekly traffic volume. This model will be based on the random forest algorithm. Additionally, a "competing" model based on linear regression will be developed. The aim is to compare the effectiveness of both methods in reflecting the nature of the analyzed traffic. Both models are to be optimized to achieve the best possible fit. The novelty of this work stems from its methodological integration: the transformation of time variables using Fourier series to encode temporal cycles, the fine-tuned comparison of model residuals using distributional statistics, and the systematic evaluation of model training time versus performance trade-offs, a perspective that is often underreported in related studies. In summary, this research fills a methodological and applied gap in the current literature by delivering a reproducible, scalable, and empirically validated modeling framework for short-term traffic prediction. It offers insights into the efficacy and practical limitations of machine learning models relative to simpler baselines, ultimately guiding future deployments in urban mobility planning. Further objectives include evaluating the quality of both models using the following metrics: coefficient of determination R^2, adjusted R^2, mean of the residuals' distribution, standard deviation of the residuals' distribution, kurtosis of the residuals' distribution, normality of the residuals' distribution, mean and standard deviation of the R^2 distribution on random samples and average model construction time [16]. The main programming environment used in this study is R [17].

To sum up this article, compares two approaches to traffic modeling: a linear trigonometric regression model and a nonlinear random forest model. The novelty lies in the optimized implementation of the linear model, which enables fast and resource-efficient generation of forecasts, and the author's parameter selection algorithm for the random forest model, which takes into account the seasonality and variability of the data. The in-house contribution includes a detailed comparative analysis of the two models in terms of the quality of fit, the distribution of residuals, the stability of the R^2 coefficient of determination, and the effect of the number of input variables on the model-building time. The experiments conducted showed that random forest achieves high prediction quality already with a limited number of variables, making it an effective and flexible tool for analyzing seasonal data, such as traffic volume.

The paper is divided into five sections. The introduction provides a review of the literature and recent trends in traffic prediction optimization. "General description of the used dataset" offers some information on the dataset and its preprocessing. "The modelling process" describes the models applied: linear regression and random forest, as well as their application in this study. "Model testing" discusses model evaluation, including residual distribution and R-squared values on new data. The final section provides a summary, conclusions, and outlines the scope of future research.

GENERAL DESCRIPTION OF THE USED DATASET

The study utilized data obtained from the Municipal Road Authority in Rzeszów. The dataset contains information regarding the number of vehicles passing through a network of measurement points distributed across 71 intersections and pedestrian crossings in Rzeszów. The measurements were taken in February and July 2022 and were recorded at hourly intervals. The dataset includes 1.416 measurements collected at each of the 399 measurement points, resulting in a total of 564,984 data points.

The original data is in the form of 59 Excel files, one for each day. Each worksheet in the file is dedicated to an individual intersection or pedestrian crossing. In Table 1, one can see a fragment

Table 1. The worksheet contains data on traffic flow at the intersection of Powstańców Warszawy Avenue and Batalionów Chłopskich Avenue on 1.07.20224

Powstańców Warszawy Avenue - Batalionów Chłopskich Avenue												
Time\Inlet		North - from Dąbrowskiego Street			East - from Powstańców Warszawy Avenue		South - od from Podkarpacka Street		West - from Batalionów Chłopskich Avenue			Total
		Relation			Relation		Relation		Relation			
		left	ahead	right	left, ahead	right	left, ahead	right	left	ahead	right	
00:00:00	01:00:00	14	85	11	159	11	69	61	11	80	17	518
01:00:00	02:00:00	5	39	5	87	5	68	39	9	70	18	345
02:00:00	03:00:00	7	30	6	76	2	61	32	7	58	11	290
03:00:00	04:00:00	6	39	3	85	3	100	38	4	43	20	341
04:00:00	05:00:00	13	73	10	135	6	153	68	3	79	42	582
05:00:00	06:00:00	17	165	25	485	25	374	200	8	188	81	1568
06:00:00	07:00:00	54	264	67	1068	52	665	398	36	489	279	3372
07:00:00	08:00:00	66	310	89	1008	61	840	462	83	661	235	3815
08:00:00	09:00:00	83	348	74	962	73	736	428	79	557	242	3582
09:00:00	10:00:00	75	378	76	915	83	666	486	86	740	233	3738
10:00:00	11:00:00	108	357	72	1146	77	657	470	77	742	264	3970
11:00:00	12:00:00	81	415	71	1219	101	674	475	66	614	237	3953
12:00:00	13:00:00	91	441	74	1241	68	717	448	72	714	240	4106
13:00:00	14:00:00	71	493	74	1155	69	713	483	88	658	263	4067
14:00:00	15:00:00	63	464	72	1317	69	880	377	110	593	243	4188
15:00:00	16:00:00	55	469	74	1140	71	650	282	105	543	224	3613
16:00:00	17:00:00	52	456	66	1096	58	576	376	104	579	230	3593
17:00:00	18:00:00	61	372	70	1047	52	554	456	60	703	222	3597
18:00:00	19:00:00	51	356	49	1130	61	485	464	53	684	193	3526
19:00:00	20:00:00	75	338	53	1027	37	498	466	57	550	178	3279
20:00:00	21:00:00	52	261	37	917	52	397	334	43	502	144	2739
21:00:00	22:00:00	42	226	31	763	42	346	236	35	396	103	2220
22:00:00	23:00:00	32	189	26	558	34	298	228	27	260	64	1716
23:00:00	00:00:00	21	159	24	289	19	161	135	17	186	33	1044
Total		1195	6727	1159	19025	1131	11338	7442	1240	10689	3816	63762
Date: 01.07.2022			Updated: 06.12.2023 12:28:41									

of the worksheet containing data on traffic flow at the intersection of Powstańców Warszawy Avenue and Batalionów Chłopskich Avenue on 1.07.2022.

To facilitate further processing, data from selected intersections were copied into a new file in the form of a uniform table (Table 2), where the rows correspond to consecutive measurements, and the columns describe the time of measurement and the recorded values at individual connections. At this stage, the connections were also assigned abbreviated names, which are explained in the next paragraph.

One of the time measures in the table above is weekly hours. Week hour equal to 1 corresponds to the time interval from 0:00 to 1:00 on Monday, week hour equal to 2 corresponds to the interval from 1:00 to 2:00 on Monday, and this continues up to 168, which corresponds to the interval from 23:00 on Sunday to 0:00 at the transition from Sunday to the following Monday.

Using weekly hours as an explanatory variable allowed for the development of a model fitted to weekly traffic intensity, which is in line with the objective of the study.

Figure 1 clearly shows distinct daily cycles, with increased traffic during the day and reduced traffic at night, as well as local peak hours. Typically, traffic is noticeably lower on weekends. A recurring weekly cycle is also evident. This cycle is somewhat blurred for certain connections, such as *Ht1_Ldl*, due to factors not directly related to time. However, the majority of connections

Table 2. A set of processed tables with data prepared for import to the R environment

Date	Week hour	Db_PW	Db_Pk	…	BCh_Pk
1.07.2022	121	14	85	…	17
1.07.2022	122	5	39	…	18
…	…	…	…	…	…
1.07.2022	129	83	348	…	242
…	…	…	…	…	…

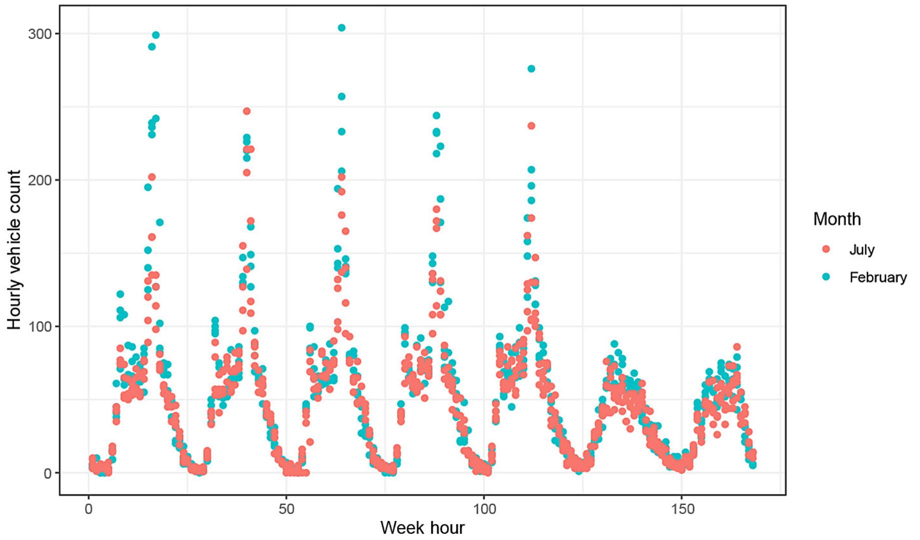

Figure 1. Registered traffic volume on the *BCh_Db* connection over a weekly span. The traffic is visibly higher in February than in July

exhibit a high concentration of measurement points around a specific curve, which is promising for a model based on a weekly perspective.

It was noted that on some connections, the traffic was different between the two tested months, most notably in the case of the *BCh_Db* connection, where the traffic in February was on average 15% higher than in July. It is clear that seasonal aura has some impact on the traffic volume, but to test this notion further, a more expansive dataset is required to cover the other 10 months as well. Due to constraints of the current dataset, this has to be tested in the future.

17 out of the 71 monitored connections were selected for the study, each with 1,416 measurement points. Each connection received its linear model and random forest model. Within the intersection of Powstańców Warszawy Avenue and Batalionów Chłopskich Avenue, the following connections were selected:

- *Db_PW* – left turn from Dąbrowskiego St. to Powstańców Warszawy Av.,
- *Db_Pk* – straight passage from Dąbrowskiego St. to Podkarpacka St.,
- *Db_BCh* – right turn from Dąbrowskiego St. to Batalionów Chłopskich Av.,
- *PW_Pk.BCh* – left turn from Powstańców Warszawy Av. to ul. Podkarpacka and straight passage to Batalionów Chłopskich Av.,
- *PW_Db* – right turn from Powstańców Warszawy Av. to Dąbrowskiego St.,
- *Pk_BCh.Db* – left turn from Podkarpacka St. to Batalionów Chłopskich Av. and straight passage to Dąbrowskiego St.,
- *Pk_PW* – right turn from Podkarpacka St. to Powstańców Warszawy Av.,
- *BCh_Db* – left turn from Batalionów Chłopskich Av. to Dąbrowskiego St.,
- *BCh_PW* – straight passage from Batalionów Chłopskich Av. to Powstańców Warszawskich Av.,
- *BCh_Pk* – right turn from Batalionów Chłopskich Av. to Podkarpacka St.
- In the intersection of ul. Hetmańska and ul. Wincentego Pola, the following connections were chosen:
- *Ht1_Ldl* – left turn from Hetmańska St. (north inlet) to the Lidl store parking lot,
- *Ht1_Ht2* – straight passage through Hetmańska St. (from north to south),
- *Ht1_WP* – right turn from Hetmańska St. (north inlet) to Wincentego Pola St.,
- *Ldl_Ht1.WP.Ht2* – exit from the Lidl store parking lot in any other direction,
- *Ht2_Ht1.WP* – straight passage through Hetmańska St. (from south to north) and left turn to Wincentego Pola St.,
- *Ht2_Ldl* – right turn from Hetmańska St. (south inlet) to the Lidl store parking lot,
- *WP_Ht1.Ldl.Ht2* – passage from Wincentego Pola St. to any other direction.

It should be noted that combining the individual models of connections in a similar manner to what is done in [9, 10] is not feasible because not all connections are “one-to-one”. There are also “one-to-many” connections, like *WP_Ht1.Ldl.Ht2*, which cannot function as an edge in a graph.

THE MODELLING PROCESS

Since the study focuses on modeling traffic intensity weekly, the primary explanatory variable for each of the considered models is the hour of the week in which each measurement was taken. The range of this variable is from 1 to 168, meaning the resolution of the model (the length of the model cycle) is 168. Following the trigonometric model approach, the primary explanatory variable was transformed into a series of pairs of normalized sines and cosines, functionally similar to a Fourier series [11]. To achieve this, the following script was used (see Listing 1):

The first pair of curves has a period of 168 hours, completing one full cycle within a week. The next pair has a period of 84 hours, thus completing two full cycles in a week, and so on. Each subsequent pair completes one more cycle than the previous one. The last pair completes 84 cycles in a week. The number of cycles is equal to half the number of measurements taken in a week. Tests have shown that for the linear model, further increasing the frequency of the curves does not lead to an increase in the R-squared value, indicating that it does not contribute any additional informational gain. Therefore, it can be assumed that the upper limit for the significant frequency of curves in a single model cycle is equal to half the resolution of the model.

Up to the point of reaching this limit, the goodness-of-fit of the linear model (measured by the R-squared coefficient) increases with each additional pair of curves as model variables. In contrast, in the case of the random forest model, the number of pairs of curves does not significantly affect the model’s goodness-of-fit. The changes in the R-squared values based on the number of pairs of curves were investigated separately for each connection considered. Figures 2 and 3 present some of the notable examples.

A noticeable stepped shape of the fit graph of the linear model can be noted. This is observed in all studied connections. The jumps typically occur at points where the number of curve pairs is a multiple of 7. This is because the weekly model cycle consists of 7 daily cycles that are generally very similar to one another. An interesting exception is the connections where the daily cycles differ. In such cases, the traffic pattern on weekends is often different from that on weekdays. The best example of such a connection is *Ht1_Ldl*.

This observation leads to the conclusion that for most of the studied connections, a linear model can be constructed using only curves with frequencies that are multiples of seven, with only a slight decrease in R-squared. This is because these frequencies provide the greatest informational gain. This approach would allow for a somewhat “slimmer” linear model at the cost of slightly worse goodness-of-fit.

Listing 1. A function for transforming the primary explanatory variable

```
fourierize <- function(t, range) { #t = week hour, #range = vector of sine/cosine frequencies to be used
  result <- c()
  for (i in range) {
    result <- c(result, sin(2*t*i*pi/168)) #sine function value
  }
  for (i in range) {
    result <- c(result, cos(2*t*i*pi/168)) #cosine function value
  }
  return(result)
}
```

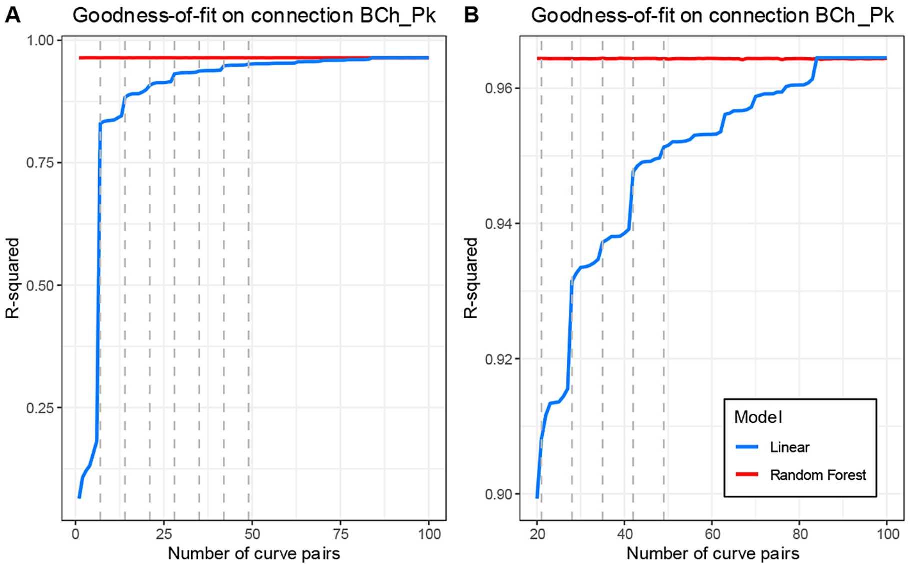

Figure 2. The R-squared value as a function of the number of curve pairs on connection *BCh_Pk*. Chart A presents the full range, while Chart B is zoomed in on the further part of the range, demonstrating that the stepped pattern still occurs. Dashed lines represent selected multiples of 7

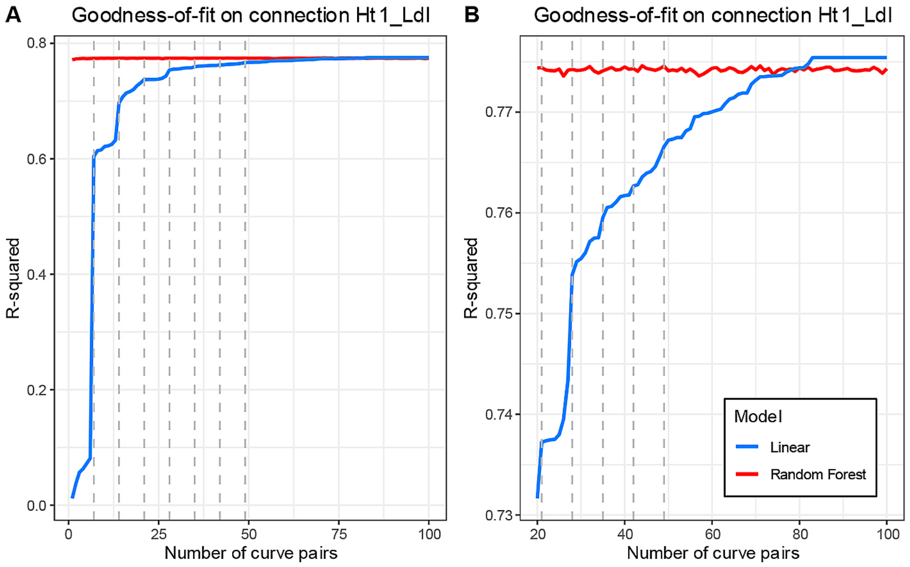

Figure 3. The same type of graph as Figure 2, this time the connection presented is *Ht1_Ldl*. The stepped pattern is less prevalent and is visibly decaying with each new curve pair

In most cases, the goodness-of-fit levels for both types of models are quite similar, although the linear model generally has an R-squared value that is a few thousandths higher, which is not a significant difference. A key conclusion from the analysis of the above graphs is that the random forest model does not require an elaborate set of explanatory variables in the form of sines and cosines to achieve the highest possible fit on any of the studied connections. This simplification allows for an easier modeling process for this type of model.

BUILDING THE LINEAR MODEL

The construction of the linear model follows a process similar to that of the trigonometric model described in [9, 10]. The model consists of a regression equation (common to all connections) and a coefficient table that stores an individual set of regression coefficients for each connection. Given that the explanatory variables are a series of sines and cosines with frequencies ranging from 1 to 84, the general form of the regression equation is as follows:

$$f(t) = \beta_0 + \sum_{i=1}^{84}(\beta_i \sin\frac{ti\pi}{84} + \beta_{i+84}\cos\frac{ti\pi}{84}) \quad (1)$$

where: β_0, β_1, β_2,… are the regression coefficients, t - is the current time expressed in week hours.

Filling in the coefficient table is an automated process that utilizes built-in R functions (Table 3). Having prepared the regression equation and the coefficient table, we can make predictions by using the following script – Listing 2.

The first line of this function (see Listing 2) assigns the value of the intercept (aka β_0) to the result, and then sequentially adds the value of each trigonometric function multiplied by its corresponding coefficient. Using the random forest package [17], the function somewhat "manually" performs the work of the built-in *predict* function in the R environment, but does so in a more optimized manner. Also, storing the coefficient table and the *LM_predict* and *fourierize* functions requires less memory than storing a whole series of models, since each connection has its own model. This makes this approach highly optimized in computational terms.

BUILDING THE RANDOM FOREST MODEL

When constructing the random forest model, the selection of two input parameters is crucial: *ntree*, which represents the number of decision trees in the model, and *nodesize*, which indicates the size of the random subset of variables at each node. To achieve this, a grid search algorithm was employed to test all combinations of these two parameters and to identify the one that provides the highest R-squared value. It is important to note that the R-squared coefficient used at this stage comes from the internal validation tests of the random forest algorithm and may slightly differ from the R-squared obtained at other stages of the research [17].

The range of values tested in the case of *ntree* parameter ranges from 10 to 40, while for the *nodesize* parameter, it is from 5 to 20. The results of the algorithm's execution were recorded, and

Table 3. Fragment of the coefficient table

Connection	β_0	β_1	β_2	β_3	…
Db_PW	45.85	3.13	4.63	2.58	…
Db_Pk	246.32	19.98	33.92	15.86	…
Db_BCh	43.38	5.88	6.96	1.87	…
PW_Pk.BCh	712.35	74.60	89.74	46.77	…
PW_Db	43.90	3.96	4.64	1.36	…
Pk_BCh.Db	430.55	62.50	63.32	24.25	…
…	…	…	…	…	…

Listing 2. Fragment of the linear model prediction script

```
LM_predict <- function(t, conn) { #t = week hour, #conn = connection
  result <- LM_params$beta0[conn] #initiating the result variable with the intercept value
  for (i in 1:167) {
    result <- result + LM_params[conn,2+i]*fourierize(t,1:84)[i] #adding coefficients multiplied by their respective sine/cosine value to the result
  }
  return(result)
}
```

based on them, a data frame was created to store the optimal parameter values for each connection. The entire script is as follows (see Listing 3):

The seed is set so that random forest ‘s randomness does not affect the results. The results of this algorithm are best displayed on a heatmap in Figure 4. The analysis of the heatmaps provides several observations. First, manipulating the parameters *ntree* and *nodesize* induces very minimal reactions. The range of the R-squared coefficient is at most 0.04, and in most cases, it is significantly smaller, on the order of thousandths. There is a fairly significant correlation (r = -0.63) between the average value of the R-squared coefficient and its range. This suggests that poorly fitted models are more susceptible to manipulation of the *ntree* and *nodesize* parameters, although the sample of 17 connections studied in this work is too small to draw definitive conclusions on this matter.

Another observation is the overall increase in the R-squared value with an increasing number of trees in the model. Although this trend is present across all studied connections, it is not significantly correlated (r = -0.13) with the R-squared value in the optima. The correlation level between the values of the *ntree* and *nodesize* parameters in the optima was also examined. It amounted to 0.26, which does not indicate a significant relationship between the values of these parameters.

In conclusion, given the current sample size, it is not possible to determine definitive patterns in the location of the optima while manipulating

Listing 3. Grid search algorithm for optimizing random forest parameters

```
for (nodesize in 5:20) { #loop iterated by nodesize parameter
  for (ntree in 10:40) { #loop iterated by ntree parameter
    traffic_RF[nrow(traffic_RF), 1:2] <- c(nodesize, ntree) #adding parameter values to the result dataframe
    for (conn in 44:ncol(traffic_main)) { #loop iterated by connection
      RF <- randomForest( #building the model with the given parameter set
        x = traffic_main[,6:43], #predictor variables
        y = traffic_main[,conn], #response variable
        nodesize = nodesize,
        ntree = ntree
      )
      traffic_RF[nrow(traffic_RF), conn-41] <- mean(RF$rsq) #adding the R-squared value for a specific connection to the dataframe
    }
    traffic_RF[nrow(traffic_RF)+1, 1] <- NA
    print(paste(‘nodesize = ’, nodesize, ‘, ntree = ’, ntree, sep = ‘’)) #progress indicator
  }
}

traffic_RF$mean <- rowMeans(traffic_RF[,3:(ncol(traffic_RF)-1)]) #adding the average of 17 connections for each parameter set to the dataframe
```

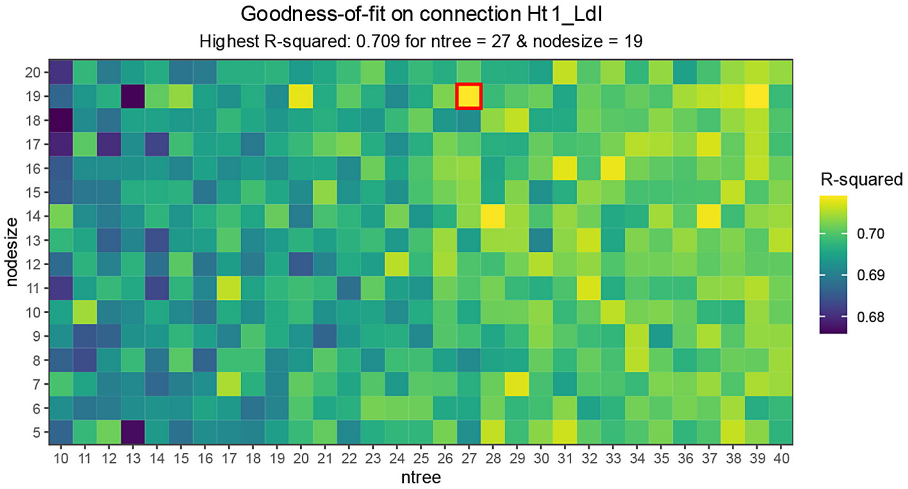

Figure 4. Heatmap illustrating the results of the grid search for *Ht1_Ldl* connection. The optimum is marked in red

the *ntree* and *nodesize* parameters. It can also be stated with moderate certainty that the appropriate selection of these parameters is more crucial in models with a lower level of fit. The optimal parameters for the model established during the tests were saved in a data frame. Individual parameters will be used in the model construction for each connection.

MODEL TESTING

The primary goal of any model is to appropriately fit the data set, which is why a series of plots was used to visualize the fit of both types of models across each of the studied connections. The actual measurement points are represented by gray dots, while the lines indicate the values predicted by the models. The linear model is marked in blue, and the random forest model is marked in red (Figures 5–6).

To verify if the predictions of both models are identical, the Wilcoxon signed-rank test [18] with a significance level of 0.05 was used as the populations were not expected to follow the normal distribution. The null hypothesis is that predictions of both models are identical. Each connection was tested separately. The results are inconclusive, as for 8/17 connections the p-value is below the significance level, indicating that for

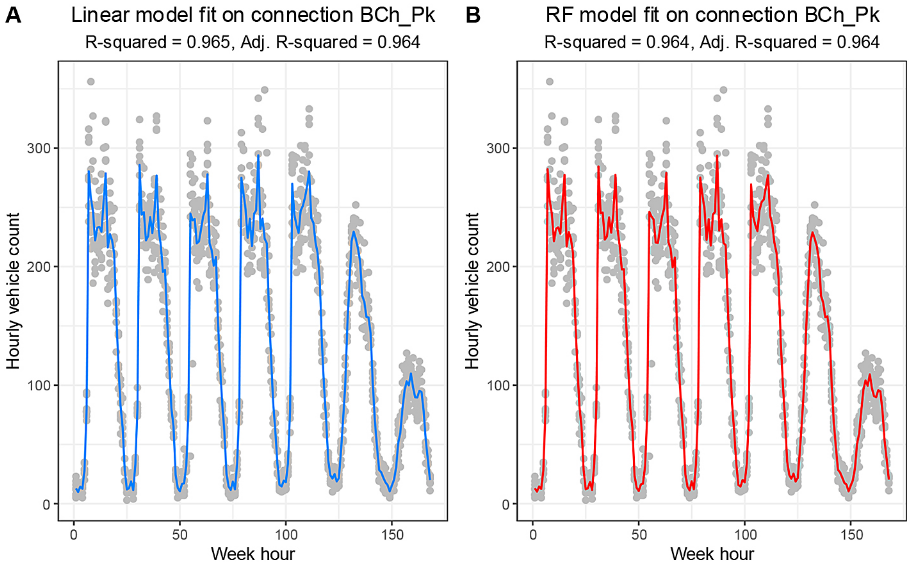

Figure 5. Fit of both tested models on *BCh_Pk* connection, where the measuring points scatter is very minimal. There are next to no visible differences

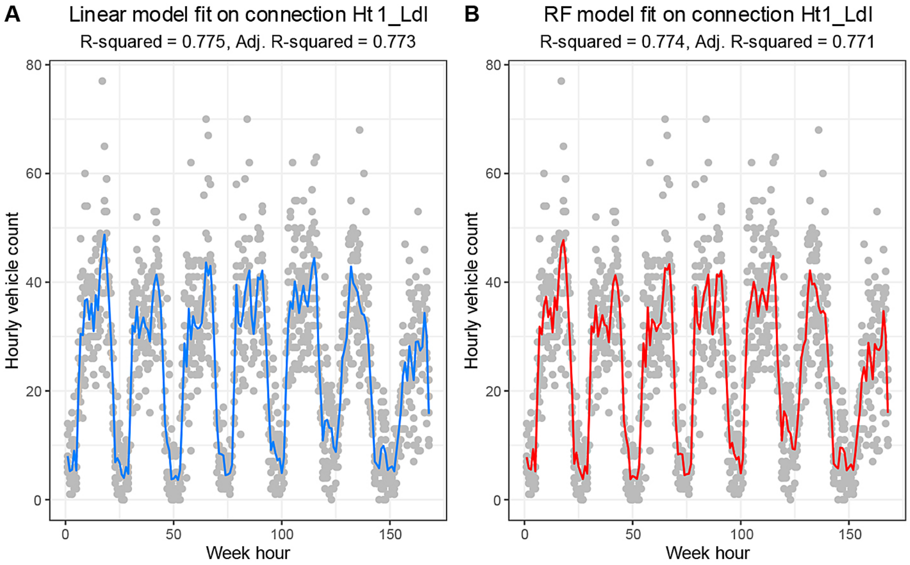

Figure 6. Fit of both tested models on the *Ht1_Ldl* connection, where the scatter is wider. While still very similar, there are some more pronounced differences, e.g., around the dip in the vicinity of week hour 120, corresponding to Friday night

these connections the predictions are significantly different. For the remaining connections, there is no basis to reject the null hypothesis.

Visually, any differences between the models are very difficult to spot, even in the case of connections where both models were marked as significantly different by the Wilcoxon test. The lowest p-value was noted in the *Ht2_Ht1.WP* connection. However, R-squared for this connection is 0.976 for both models; therefore, by this metric, the models are not that different since they are both very close to the perfect fit. The differences between each model and the actual values were also tested. In the case of the linear model, no significant differences were identified by the Wilcoxon test. For the random forest model, only 1 connection notes a significant difference, that being the same connection as mentioned in the paragraph above. This contrasts with the near-perfect R-squared value; therefore, further tests are necessary to properly assess the model's performance.

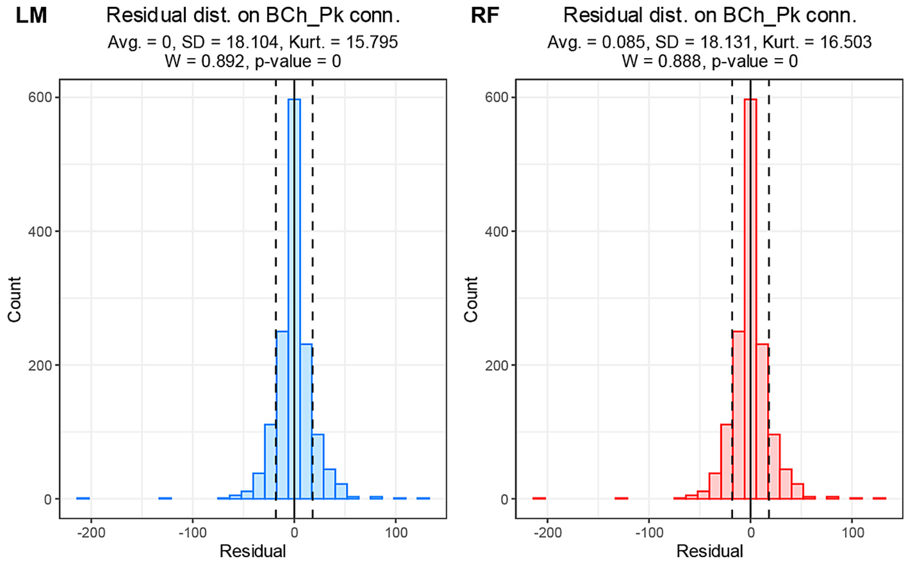

Figure 7. Residual distribution on the *BCh_Pk* connection. Note the tall central spike indicating a near-perfect fit in most measuring points. Accordingly, the kurtosis is very high (around 16 in both cases). Several outliers are visible, representing traffic anomalies

RESIDUAL DISTRIBUTION

Another indicator of model quality is the distribution of residuals. The general aim for it is to be as close as possible to a normal distribution. The linear model is marked in blue, while the random forest model is marked in red. Additionally, the plots include the mean, standard deviation, and kurtosis of the distribution, as well as the results of the Shapiro-Wilk normality test. All these values are rounded to three decimal places (see Figures 7–8).

The most conclusive observation from this testing is the p-value close to zero. This provides grounds to reject the hypothesis of normality of the distributions. The most likely cause of such test result is positive kurtosis of the distributions, indicating that the distribution is more peaked than a normal distribution. With such a large sample (1416 measurement points), even a small deviation of kurtosis from zero drastically lowers the p-value. In some cases, such high kurtosis might suggest an overfit of the models; however, the model fit test on new data discussed in the next section dispels these doubts.

Two things should be noted. First, with a large sample size, the sensitivity of the Shapiro-Wilk test increases, causing a distribution that only slightly deviates from normal to be interpreted as strongly different from normal [19]. Analysis of histograms and kurtosis leads to the conclusion that some of the examined residual distributions deviate just barely from normality, as they are roughly symmetrical, and the kurtosis in several cases is close to 1.

Second, the deviations of the residual distributions from normality are not, in this case, indicators of significant flaws in the model due to the large sample size [20]. If a smaller sample were taken from the residual distribution, the p-value would probably rise above the significance level (we could assume a typical α = 0.05), which would change the outcome of the test and allow the distribution to be considered normal.

R-SQUARED DISTRIBUTION ON NEW DATA

To test the fit of the models on new data, the dataset was split into a training subset and a test subset in proportions of 80% and 20%, respectively. Then, the R-squared value obtained by the model on the test set was measured. This entire process was repeated 1000 times with different subsets for each connection and each type of model separately, setting the seed sequentially from 1 to 1000 to eliminate randomness as a factor [21].

An interesting observation is the left (negative) skewness of all the above distributions, typically stronger in better-fitting connections (i.e., those with higher average R-squared values). The coefficient of skewness with the largest absolute value (approximately -0.94 for the linear model and -0.93 for the random forest model) is found in

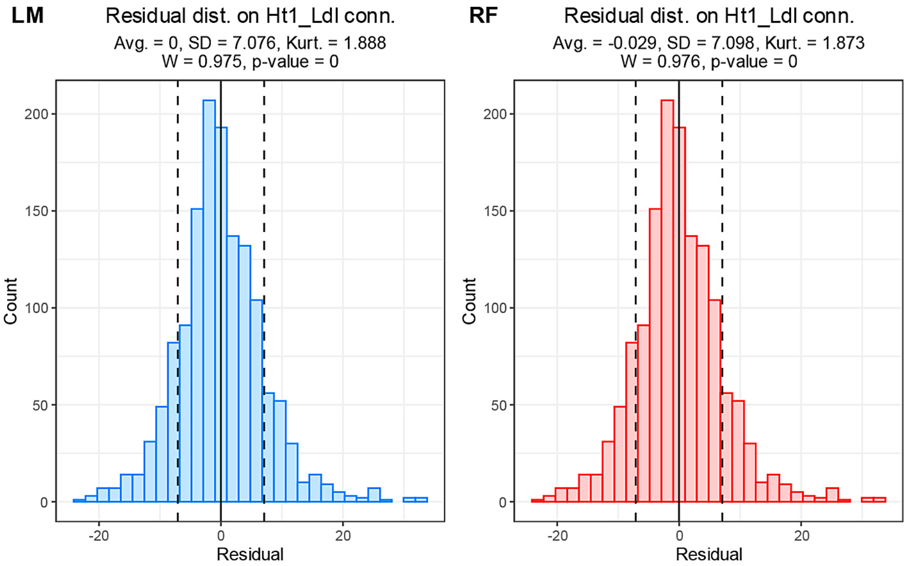

Figure 8. Residual distribution of both tested models on the *Ht1_Ldl* connection. This time, there is no narrow spike at the center. The distribution is less concentrated as the model cannot fit closely to the measuring points scattered so widely. In line with this observation, the kurtosis is much lower at around 1.9 in both cases

the connection *PW_Pk.BCh*. On the other hand, the connection *Ht1_Ht2* exhibits the smallest absolute value of the skewness coefficient (approximately -0.1 for the linear model and -0.11 for the random forest model).

There is a moderate correlation between the skewness coefficient and the average value of the R-squared distribution (see Figure 9–10). It is approximately -0.41 for the linear model and -0.413 for the random forest model. This indicates a relationship between the skewness of the R-squared distribution and its average value. Specifically, the higher the model fit on a specific connection (i.e., higher average R-squared), the fewer outlier values on the right side of the distribution and more is on the left. This suggests that the model is approaching the limits of its fitting capability, as even "lucky" splits into the training and test subsets do not significantly raise the score.

However, there are exceptions to this rule, such as the connection *Ht1_Ht2*, which shows one of the highest average R-squared values while having a skewness coefficient almost equal to zero, indicating it is nearly symmetrical.

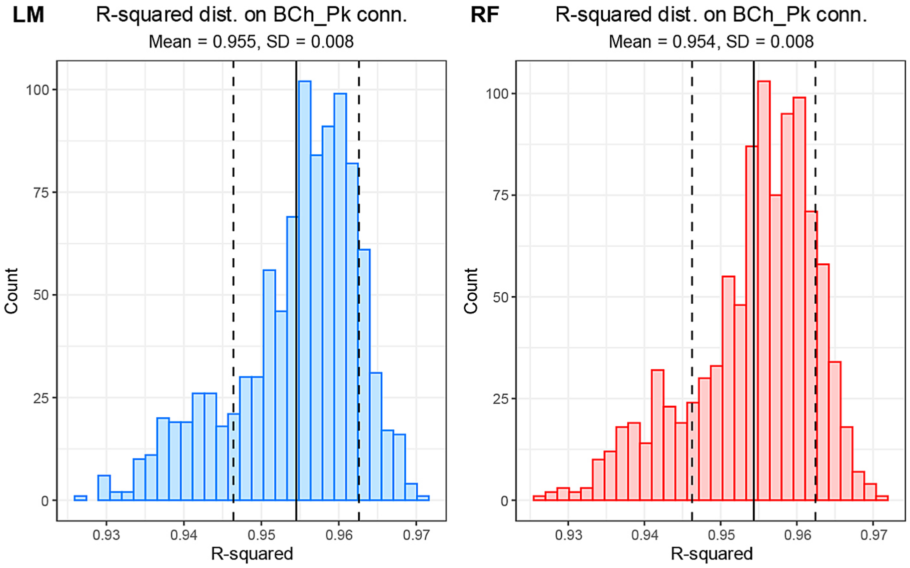

Figure 9. R-squared distribution of both tested models on the *BCh_Pk* connection. Note the left (negative) skewness

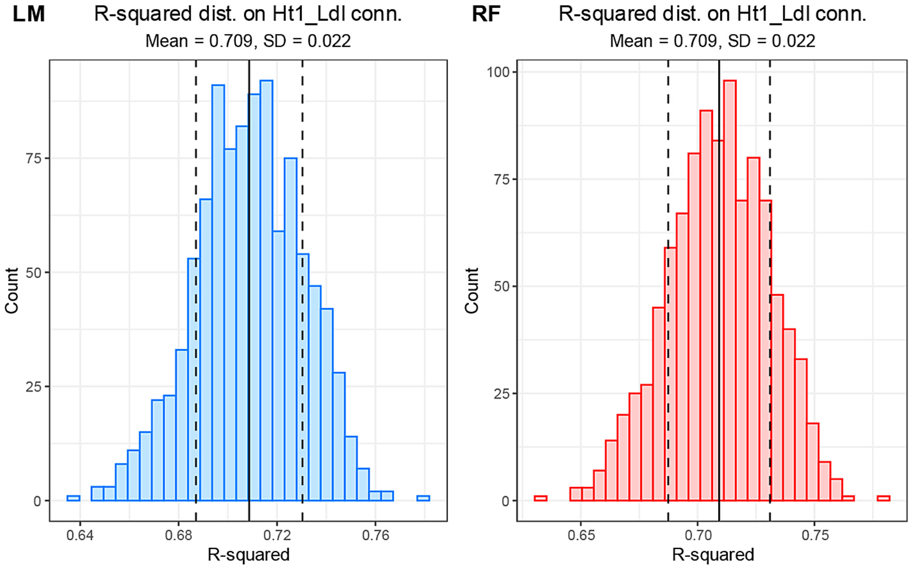

Figure 10. R-squared distribution of both tested models on the *BCh_Pk* connection. No obvious skewness can be discerned

To follow up on the significant difference in predictions made by the random forest model in connection *Ht2_Ht1. WP,* as indicated by the Wilcoxon test, special attention was paid to its performance in other tests. However, no deficiencies were identified to corroborate the result of the Wilcoxon test; therefore, the model's performance is deemed satisfactory.

A Wilcoxon test was also performed on the R-squared distribution on the new data for each connection separately. First, a two-sided test was performed to identify any significant differences. At a significance level of 0.05 the null hypothesis was rejected for all connections but *Db_Pk*, meaning there are significant differences at 16/17 tested connections. A two-sided test was followed up with a right-sided test with the intention to verify if the random forest model has a significantly higher R-squared in this experiment. This hypothesis was rejected in 14/17 connections. For the sake of completeness, an opposite, left-sided test was also performed, resulting in rejection of the null hypothesis of the random forest model having a lower R-squared in 2/17 connections. Table 4 presents a summary of the obtained Wilcox test results.

Overall, in this test, the linear model was more favorable in 82% of the tested connections. This suggests an overwhelming advantage of the linear model in predicting values it was not trained on, but in truth, the absolute differences of average R-squared values on each connection are minimal, reaching the third decimal place at best. These results should ideally be verified with a sample larger than 17 connections to confirm whether the linear model advantage will hold in a larger population.

MODEL BUILDING TIME

An important issue from a practical perspective is the computation time required for model training. Due to the use of two completely different algorithms, the linear model and the random forest model have different time demands.

The selection of explanatory variables can have a significant impact on model-building time, so this factor was examined first. The time was measured by sequentially adding another sine/cosine pair to the model. Measurements were taken once for each connection, and then the average for a specific number of curve pairs was calculated (see Figure 11).

A clear upward trend is evident in both types of models, indicating that the introduction of additional explanatory variables increases the time required for their construction. In the case of the random forest model, this trend is almost perfectly linear (the correlation between the number of

Table 4. Summary of paired Wilcox test results at a significance level of 0.05. Each cell presents a p-value for a specific test at a specific connection. The results are rounded to the third decimal place

Connection	Two-sided test	Right-sided test	Left-sided test	Verdict
Db_PW	0.000	0.000	1.000	Linear model preferred
Db_Pk	0.739	0.369	0.631	Both tied
Db_BCh	0.000	0.000	1.000	Linear model preferred
PW_Pk.BCh	0.000	0.000	1.000	Linear model preferred
PW_Db	0.000	0.000	1.000	Linear model preferred
Pk_BCh.Db	0.004	0.998	0.002	Random Forest preferred
Pk_PW	0.000	0.000	1.000	Linear model preferred
BCh_Db	0.000	0.000	1.000	Linear model preferred
BCh_PW	0.000	0.000	1.000	Linear model preferred
BCh_Pk	0.000	0.000	1.000	Linear model preferred
Ht1_Ldl	0.000	1.000	0.000	Random Forest preferred
Ht1_Ht2	0.000	0.000	1.000	Linear model preferred
Ht1_WP	0.000	0.000	1.000	Linear model preferred
Ldl_Ht1.WP.Ht2	0.000	0.000	1.000	Linear model preferred
Ht2_Ht1.WP	0.000	0.000	1.000	Linear model preferred
Ht2_Ldl	0.001	0.000	1.000	Linear model preferred
WP_Ht1.Ldl.Ht2	0.000	0.000	1.000	Linear model preferred

pairs of functions and time is close to one). For the linear model, it appears to be a power trend, as the correlation is about 0.95, but the correlation between the square of the number of pairs of functions and time is approximately 0.98.

Significantly longer average execution time for the random forest model across the entire range studied is also observed. Additionally, the upward trend is stronger for this type of model. However, since the linear model exhibits a power growth trend, it could become more time-consuming to construct with a sufficiently large number of explanatory variables. This number, however, is impractically large in the context of the research documented in this work.

Despite the fact that the random forest model is more time-consuming than the linear model when both have the same number of variables, it has a property that completely reverses this situation in a broader picture. Specifically, the random forest model does not require such a large number of curve pairs to achieve its maximum performance (see Figures 2–3). It only needs 1 curve pair to reach a fitting level nearly equal to that of the linear model with 84 curve pairs. The random forest model using 1 curve pair requires, on average, about 0.021 seconds for construction, while the linear model using 84 pairs takes about 0.039 seconds, which is nearly twice as long. This means that the random forest model is more time-efficient while maintaining a very similar level of fit. The dependence of the random forest model's construction time on the parameters *ntree* and *nodesize* was also examined Figure 12). As before, the time was averaged across all 17 connections. For this test, a model using 1 curve pair was employed.

The time increases with the number of trees in the model, while its dependence on the *nodesize* parameter is unclear. The heat map (Figure 12) is divided into distinct horizontal bands, creating two alternating groups. This is not an ideal division, as the bands appear to be slightly "skewed". Additionally, there are a few outlier values present on the plot. The most prominent are at the coordinates (17, 10) and (29, 19).

If we compare Figure 12 to Figure 4, it can be observed that the fastest model has a poor fit compared to models with more accuracy-optimized parameters. This leads to the conclusion that the most accurate model is not necessarily the fastest model in the case of random forest. Depending on whether the user's needs are more focused on the time or precision of the model, a compromise can be achieved by selecting model parameters that ensure both satisfactory fit and construction time. However, it is worth noting that the time differences would become more significant if a considerably larger number of models were constructed or if models were trained on a larger dataset than considered in this work. A similar compromise can be achieved for the linear model by appropriately selecting explanatory variables, e.g., using only curves with frequencies that are multiples of seven as mentioned in "The modelling process"

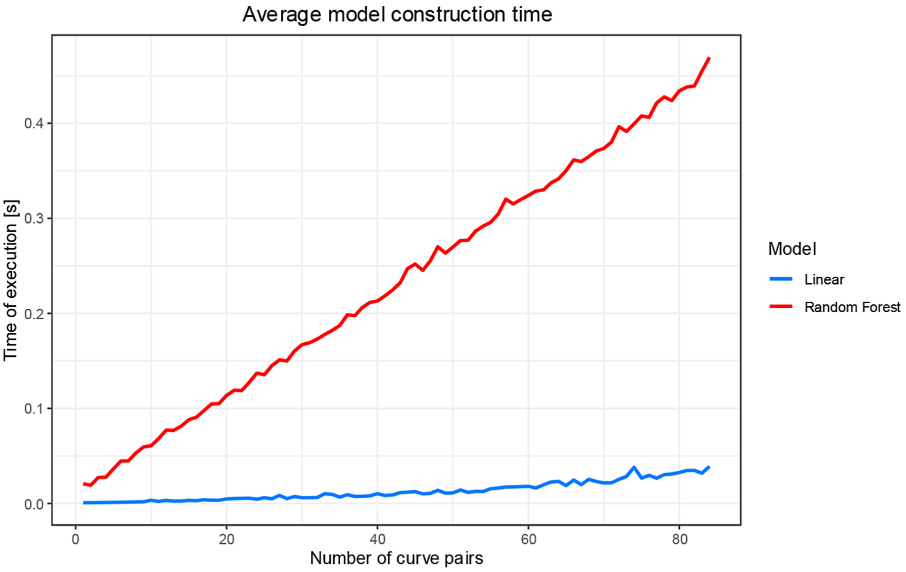

Figure 11. Dependence of the model construction time on the number of curve pairs used as explanatory variables

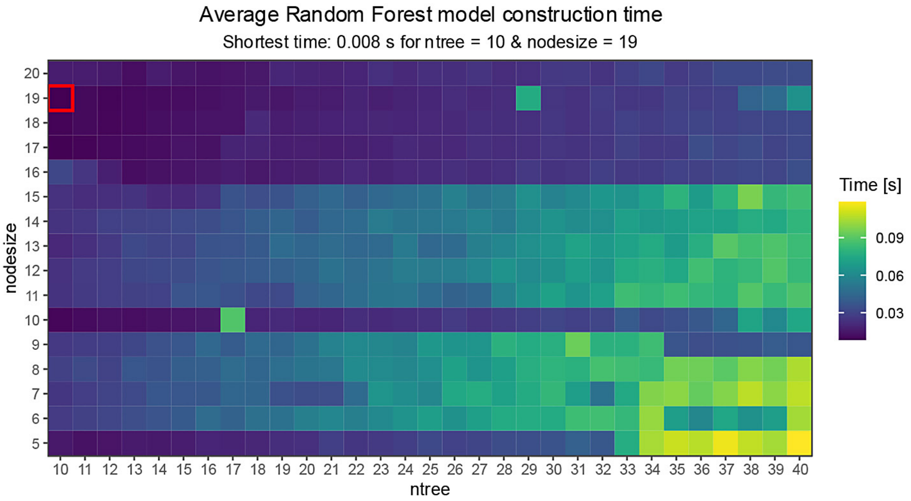

Figure 12. Dependence of the model construction time on the *ntree* and *nodesize* parameters. The least time-consuming combination is marked in red

Table 5. Summary of tested metrics. All figures represent an average across the 17 tested connections. Construction time was measured once for each connection

Metrics	Related figures	Random forest model result	Linear model result
R-squared on the full set	5, 6	0.9322	0.9329
Residual Standard Error	7, 8	24.5645	24.4694
R-squared on new data	9, 10	0.9128	0.9132
R-squared standard deviation	9, 10	0.0124	0.0122
Construction time	11, 12	0.021 s	0.039 s

section, or using Hellwig's method of selection of variables [22].

The parallel development of the linear model allowed for continuous mutual improvement of both types of models. As a result, a fitting as good as the available dataset allows was achieved for both types, which was confirmed by a series of tests (Table 5).

CONCLUSIONS

Throughout the research, a highly accurate and statistically significant regression model based on the random forest method was constructed, as can be seen in Table 5; the results are very close, sometimes differing only in the fourth decimal place. In almost all metrics, the linear model proves to be slightly better, although this is a minimal advantage. This conclusion is supported by the results of Wilcoxon test performed on the R-squared on new data. In terms of sheer numbers, the only advantage of the random forest model is the construction time, which was found to be shorter by half, provided that the input variables were optimally selected.

From a technical standpoint, both models have their strengths and weaknesses. The advantages of the random forest model over the linear model include: no need for special pre-processing of the dataset before training (transforming it into a series of sine and cosine functions), shorter model construction time while maintaining the same level of fit, and easier tuning of model parameters. Conversely, the advantages of the linear model include: slightly better fit with proper data preprocessing, continuity and cyclicity of the regression curve (allowing for its infinite extension without introducing points of discontinuity), smaller memory requirements, easier handling, and more built-in quality metrics.

The model based on the chosen machine learning method proved to be almost as good as the most refined linear regression model. The differences between these types of models are more about technical aspects and user experiences, so the choice of one method depends on user preferences. Both models show significant potential in traffic prediction and analysis, supporting decision making regarding road network expansion and maintenance, and, with appropriate modifications, in other fields.

This study, while offering valuable insights, is based on a relatively limited dataset, which may affect the broader applicability of the findings. The results are closely tied to the specific urban context in which the data was collected, and as such, may not fully capture the diversity of conditions present in other cities or traffic systems. The analysis was based on a limited sample of traffic connections, which may affect the generalizability of the findings to larger or more diverse urban networks. Moreover, the results are context-specific and may not directly translate to environments with different traffic dynamics or infrastructure characteristics. There is also a potential risk of overfitting, particularly in data-driven models trained on smaller datasets. Despite these constraints, both evaluated models demonstrate practical advantages: the random forest model effectively captures complex, non-linear relationships, while the linear regression model offers a lightweight and interpretable alternative suitable for real-time forecasting, especially under constrained computational resources. The comparative results, supported by statistical significance testing, reveal important differences in model performance, contributing to a better understanding of their strengths and limitations. Importantly, this work lays the foundation for further research focused on improving model scalability, transferability, and integration with multimodal urban data.

In terms of future development, a natural next step is to expand the testing period to cover an entire year. This would allow the models to account for seasonal variations in traffic volume, which can be substantial—particularly in winter months, when road conditions tend to be more challenging, as noted in the “General description of the used dataset” section. Another promising direction involves enriching the feature set with external predictors, such as weather conditions, public holidays, or calendar effects. This is particularly feasible for random forest models, which require minimal preprocessing and can flexibly incorporate diverse input variables. Such additions may help address underperformance in irregular or low-volume traffic connections. A more ambitious extension would include scaling the models to cover a broader transportation network and exploring strategies for integrating multiple local models into a unified predictive framework. Beyond traffic, the modeling approach presented here is well-suited for a variety of cyclical phenomena – such as seasonal human activities, animal migration, tidal patterns, or sunspot cycles. These broader applications even open the possibility of contributing to early warning systems for events such as solar storms or tsunamis.

REFERENCES

1. Shapiro S.S., Wilk M.B. An analysis of variance test for normality (complete samples). Biometrika, 1965; 52(3/4), 591–611,. https://doi.org/10.2307/2333709
2. Zheng Z., Lee D.H., Shi Q. Short-term freeway traffic flow prediction: Bayesian combined neural network approach. Journal of Transportation Engineering, 2006; 132(2), 114–121. https://doi.org/10.1061/(ASCE)0733-947X(2006)132:2(114).
3. Sun S., Zhang C., Yu G. A Bayesian network approach to traffic flow forecasting. IEEE Transactions on Intelligent Transportation Systems, 2006; 7(1), 124–132, https://doi.org/10.1109/TITS.2006.869623
4. Min W., Wynter L. Real-time road traffic prediction with spatio-temporal correlations. Transportation Research Part C: Emerging Technologies, 2011; 19(4), 606–616., https://doi.org/10.1016/j.trc.2010.10.002
5. Wang Y., Papageorgiou M., Messmer A. Real-time freeway traffic state estimation based on extended Kalman filter: A case study. Transportation Science, 2006; 40(2), 167–181, https://doi.org/10.1287/trsc.1050.0122
6. Lv Y., Duan Y., Kang W., Li Z., Wang F.Y. Traffic flow prediction with big data: A deep learning approach. *IEEE Transactions on Intelligent Transportation Systems*, 2015; *16*(2), 865–873, https://doi.org/10.1109/TITS.2014.2345663
7. Vlahogianni E.I., Karlaftis M.G., Golias J.C. Short-term traffic forecasting: Where we are and where we’re going. *Transportation Research Part C: Emerging Technologies,* 2014; *43,* 3–19, https://doi.org/10.1016/j.trc.2014.01.005
8. Yang B., Pan S. Real-time traffic flow prediction using a hybrid deep learning model. *Information Sciences,* 2020; *512,* 353–366, https://doi.org/10.1016/j.ins.2019.10.034
9. Dymora P., Mazurek M., Jucha M. Regression Models Evaluation of Short-Term Traffic Flow Prediction. In: Dependable Computer Systems and Networks. DepCoS-RELCOMEX 2023. Lecture Notes in Networks and Systems, 2023; 737, 51–61, https://

doi.org/10.1007/978-3-031-37720-4

10. Dymora P., Mazurek M., Jucha M. Examining the possibility of short-term prediction of traffic volume in smart city control systems with the use of regression models”. In: International Journal of Electronics and Telecommunications, 2024; 70: 31–38, https://doi.org/10.24425/ijet.2023.147711
11. Liaw A., Wiener M. Classification and Regression by random forest. In: R News, 2002; 2/3, 18–22.
12. Xu X., Bai Y., Xu L., Xu Z., Zhao X. Traffic flow prediction based on random forest in severe weather conditions. Journal of Shaanxi Normal University, Natural Science Edition, 2020; 48(2), 25–31.
13. Sun S., Yan H., Lang Z. A study on traffic congestion prediction based on random forest model. Highlights in Science, Engineering and Technology, 2024; 101, 738–749.
14. Fisher R. Statistical Methods for Research Workers, Eleventh Edition – Revised. Oliver and Boyd, 1950.
15. Freedman D.A. Statistical Models: Theory and Practice, Revised Edition. Cambridge University Press, 2009.
16. Devore J.L. Probability and Statistics for Engineering and the Sciences. Cengage Learning, 2016.
17. Documentation of R package moments. https://www.rdocumentation.org/packages/moments/versions/0.14.1 (access date: 5.05.2024).
18. Documentation of Wilcox.test() function in the R package stats. https://www.rdocumentation.org/packages/stats/versions/3.6.2/topics/wilcox.test (access date: 11.07.2025).
19. Abramowitz M., Stegun I.A. Handbook of Mathematical Functions: with Formulas, Graphs, and Mathematical Tables. National Bureau of Standards, 1970.
20. Weisstein E.W. Kurtosis Excess. From MathWorld – A Wolfram Web Resource. https://mathworld.wolfram.com/KurtosisExcess.html (access date: 5.05.2024).
21. R-squared, Adjusted R-squared and Pseudo R-squared. https://timeseriesreasoning.com/contents/r-%20squared-%20adjusted-r-squared-pseudo-r-squared/ (access date: 29.11.2024).
22. Hellwig Z., On the optimal choice of predictors. Study VI in: Z. Gostkowski (ed.): Toward a system of quantitative indicators of components of human resources development. UNESCO, 1968.