Available online at www.sciencedirect.com

ScienceDirect

Transportation Research Procedia 82 (2025) 1229–1243

Transportation Research Procedia

www.elsevier.com/locate/procedia

World Conference on Transport Research - WCTR 2023 Montreal 17-21 July 2023

Investigating of machine learning's capability in enhancing traffic simulation models

B. Dammak[a,*], F. Ciari[b], A. Jaoua[a], H. Naseri[b]

[a]*Department of Industrial Engineering, National School of Engineering of Tunis, LR-OASIS, University of Tunis El Manar, Tunis, Tunisia*
[b]*Department of Civil, Geological and Mining Engineering, Polytechnique Montréal, 2500 Chem. de Polytechnique, Montréal, QC, H3T 1J4, Canada*

Abstract

The development of agent-based modeling in traffic simulation allows for the modeling of traveler movement and decision making using predefined rules and variables. Nonetheless, the computational cost of agent-based modeling is high, and it takes a long time to generate new scenarios using these models. To address this, this study proposes a new approach to predict the results of new simulations using machine learning techniques. This paper focuses on the reproduction of the models that simulate variables reflecting traveler decision making, such as mode choice, travel distance and duration, and waiting time. A variety of data-driven techniques have been employed in this regard to model these features resulting from unanticipated activities in a dynamic environment. The proposed approach will be based on synthetic data generated from various simulation scenarios, that will be followed by a data preparation process. Therefore, the robustness of the built machine learning models was tested and assessed in different and new situations in order to evaluate their capability to reproduce the models responsible for generating the stated variables. Experiments show that the suggested solution has a high level of robustness, implying that it can replicate the final results of these models. Further, Extreme Gradient Boosting outperformed other machine learning techniques in terms of predicting simulation variables when comparing prediction accuracy and running time.

Keywords: Transport simulation; Machine learning; Data analysis; Agent based model

1. Introduction

Traffic simulation is essential to investigate individual decision (e.g., mode choice) impacts on the traffic network. Agent-Based Modeling (ABM) is a robust approach for traffic simulation, particularly for large-scale networks (De Souza et al., 2019). However, these models are computationally expensive, and they often suffer from long running times (Llorca et al., 2020). To prevail in this issue, the application of machine learning techniques can be a possible solution. Increased computation power and the availability of massive data sets (Big Data), are among the reasons ex-

* Corresponding author. Tel.: +216-29-760-110.
E-mail address: bessem.dammak@etudiant-enit.utm.tn

2352-1465
10.1016/j.trpro.2024.12.122

plaining the relevance of machine learning-based techniques to handle problems in transportation science (Tizghadam et al., 2019). In Sun et al. (2020) for example, it is now feasible to recognize patterns such as real-time traffic flow and individual vehicles movement under various traffic flow situations that may dramatically increase the efficiency of present transportation system operations and anticipate future trends.

Prior to implementing certain processes or solutions in the actual world, however, one should make sure that machine learning techniques are indeed capable of capturing the behavior of transportation systems. Artificial data, created with simulation, can be used for the study and assessment of alternative techniques, as well as for the validation of their application. Zheng et al. (2021) revealed numerous problems typically tackled with traffic simulations spans from minimize congestion, to enhancing road safety, from improving individual decision making to increasing surface transportation productivity and efficiency, among others. Agent-based modeling, as explained in Hunter and Kelleher (2022), is a specific simulation technique, that has been used to investigate, replicate and forecast transportation scenarios, in that emergent system-wide features represent how different agents interact with one another and with their environment. However, Bálint et al. (2022) showed that agent-based models present a few challenges, including the complexity of developing, testing, parameterizing, and validating models, which makes them non-trivial and computationally costly. Some research is looking into the use of machine learning techniques, to overcome such issues, thanks to key features such as generalization, scalability and real-time applicability.

The goal of this work is to apply machine learning to build robust and efficient models that predict specific aspects of passengers' actions under diverse experimental situations. This study examines not only the movement and decision-making of travelers, but also the capability of data-driven techniques in reproducing the models that produce mode choice, travel distance and duration, and waiting time.

This work is organized into four parts, beginning with a review of the literature, followed by an explanation of the methodology used to achieve the objectives. The third part provides a description of the simulation environment as well as all of the operations performed on the artificial data. The last part depicts the entire experimental procedure up to the analysis of the obtained results.

2. Literature reviews

According to Pell et al. (2017), traffic simulation models and software tools have been developed to model and plan traffic as well as examine various traffic control tactics. The goal of traffic modeling based on Azlan and Rohani (2018) is to properly replicate traffic as observed and measured on the street. It was created using modeller experience to incorporate mathematical models into the traffic system. Since the development of Agent-Based Modeling (ABM), there have been advancements in simulation software for large networks for traffic assignment and traffic flow modeling (Kagho et al., 2020). As explained in Hong et al. (2013), ABM and simulation have been applied in several transportation sectors, it studies personal transportation-related activities and behavior, as well as computational (or systemic) approaches for studying a collaborative and responsive transportation system with intelligence by modeling a collection of autonomous decisions made by subsystem entities known as agents. These tools focus on various tasks such as reproducing traffic conditions in an urban network, multi-modal simulation of road traffic, and development of activity-based demand simulators, among others, but only a few of them can integrate the various aspects of transport modeling together to build a fully integrated large-scale agent-based model simulation (Kagho et al., 2020).

While the flexibility of ABM allows for broad application, the complexity of real-world models that have been outlined in Sivakumar et al. (2022) may present challenges, such as a high demand on computational resources and computational time, in addition to the rules that govern an ABM, which can be difficult to abstract and formulate from experimental data. Therefore, the application of machine learning techniques can be efficient to predict the results of ABMs. Machine learning techniques can capture the non-linear relationship between the response variable (e.g., mode choice) and independent variables (e.g., socio-demographic) (Dong et al., 2022). These techniques showed to be accurate and efficient to model and predict many transportation-based parameters, such as mode choice (Sun et al., 2022), child mode choice (Naseri et al., 2022), trip duration (Poongodi et al., 2022), public transit waiting time (Chu et al., 2019), and vehicle engine choice (Naseri et al., 2023). Accordingly, researchers have begun to improve the performance of traffic simulation using machine learning techniques.

As such, Zhang et al. (2021) suggested that machine learning could be used in assisting inferring optimum, system-specific rules from agent-based models. By studying the behavioral patterns of agents, machine learning-based inference models such as reinforcement learning and supervised learning approaches can enhance sequential decision

making. This new discipline can use these strategies to augment classic agent-based schemes that translate agent action rules into an adaptive model. According to Sivakumar et al. (2022), machine learning approaches may also aid in the exploration of an ABM's complicated and highly dimensional parameter space in terms of sensitivity analysis, model robustness, and so on. This relates to the benefit of ABM in terms of producing realistic datasets for training machine learning algorithms. Antoniou and Koutsopoulos (2006) created a framework for estimating velocity based on machine learning techniques such as clustering and locally weighted regression (loess) algorithms. Antoniou et al. (2013) used machine learning approaches to create a framework for dynamic traffic status prediction. Jenelius and Koutsopoulos (2013) provided a statistical approach for urban road network journey time estimates utilizing low frequency probe car data. Zheng et al. (2013) suggested a two-level neural network structure. It is used to anticipate the acceleration of the following vehicle as well as to estimate the dynamic response time. Deep learning was utilized by Lv et al. (2015) and Huang et al. (2014) to forecast traffic flow.

Aside from the benefits of machine learning for agent-based models and traffic simulation, it has been proposed in Sivakumar et al. (2022) that supervised learning algorithms may be trained to reproduce an ABM's models, which helps calibrate the ABM and reduces the computing expenses of operating several ABM. This approach addressed the issue of complicated and nonlinear parameter interactions, as well as the extended processing time. In addition, Angione et al. (2022) provide an intriguing solution by evaluating several scenarios with artificial neural networks (ANN) and gradient boosted trees to determine the effectiveness of various machine learning models in replicating the integrated models. This strategy, according to the author, will enable more rigorous sensitivity evaluations for the models while using less processor time during calibration and simulation analysis. According to Cao (2022), a combined CNN-GRU deep learning model was constructed to replicate the rail logistics traffic speed model in order to regulate the rapid expansion of high-speed trains and manage the ongoing improvement of the road network. Furtado and Andreao (2022) investigates the effects of diverse circumstances on different receivers. Forty-six metropolitan regions (MAs) in Brazil were utilized to evaluate several potential policies. After about one million simulation runs, 11076 parameters were created for use in a random forest machine learning technique to imitate the agent-based model and demonstrate the needed robustness. In addition, Huang et al. (2022) described the profits of machine learning in transportation domains by improving agent behavior modeling and computation efficiency of simulation. Furthermore, Brearcliffe and Crooks (2021) studied advances in agent-based modeling by explicitly comparing and contrasting the influence of different machine learning approaches used in the same model on simulation outcomes.

Therefore, machine learning techniques can improve such analyses. However, many machine learning techniques exist, and it is not clear which of them can be the best option to regenerate the models that simulate variables reflecting traveler decision making. Artificial Neural Networks (ANNs) are famous computational techniques for information processing, data representation, and prediction (Naseri et al., 2020). ANNs could outperform many machine learning techniques in terms of prediction accuracy, such as logistic regression, gradient boosting decision tree (Hung et al., 2017), linear regression, and support vector machine (Naseri et al., 2020). Recently, ensemble learning techniques have been widely used in different prediction problems, and they showed superior performance than other prediction methods. These techniques generate a given number of weak learners and combine them to construct a powerful prediction technique (Naseri et al., 2021). As such, Random Forest is a powerful ensemble learning technique, which has been shown to be more accurate than various machine learning methods, e.g., Sequential Minimal Optimization for Regression, M5P, K-Nearest Neighbors (Kayadelen et al., 2022), Multiple Linear Regression (Naseri et al., 2022), decision tree, Adaptive boosting, Gaussian process classification, quadratic discriminant analysis, linear discriminant analysis, na¨ıve Bayes, support vector machine, and logistic regression (Maniruzzaman et al., 2018).

EXtreme Gradient Boosting (XGBoost) is another ensemble learning technique, which has been widely used for modeling classification and regression problems. XGBoost is famous for its speed, parallel processing, and high accuracy in complex scenarios (Chen and Guestrin., 2016), (Jeon et al., 2020). XGBoost could surpass many prediction techniques when comparing running time and prediction accuracy, such as Decision Trees (Jamal et al., 2021), Logistic Regression (Wang et al., 2019), Support Vector Machines (Nguyen-Sy et al., 2023). Therefore, Artificial Neural Networks, Random Forest, and XGBoost could be appropriate candidates for predicting simulation variables since they have been shown to be accurate prediction techniques in previous studies. However, it is not clear which of them is the most accurate technique for the mentioned prediction problem.

According to previous research, machine learning techniques could be accurately applied to traffic simulation, including estimating speed in Antoniou and Koutsopoulos (2006), estimating travel time of urban road network in

Jenelius and Koutsopoulos (2013), and even reproducing the speed model of rail logistics traffic in Cao (2022). However, to the best of the authors' knowledge, there has been no investigation to replicate the original patterns that exist in the simulation and few related works have attempted to address this issue. Our goal in this research is to essentially enhance the traffic simulation that employs agent-based models, which has certain limitations in that it is computationally expensive and more dependent on parameters to build scenarios. Thus, the approach proposed in this work is to try through machine learning to replicate the models responsible for generating a set of variables using synthetic data collected from different scenarios. The benefit of the created machine learning model is that they are data-based approaches that are more resistant to changes in reality, as opposed to model-based techniques that are more dependent on simulation parameters. Further, this research aims to compare the performance of three machine learning techniques based on running time and different performance indicators to detect the best technique for the mentioned prediction problem. This study will explore not only the robustness of the proposed solution, but also evaluating the testing time required, which may be useful in decreasing computational work if the solution is applied in a future work.

3. Methodology

This study examines the utility and feasibility of machine learning approaches to improve traffic simulation. The ability to reproduce the output of models integrated in traffic simulations, and the robustness of such an approach, is tested using simulation generated data. The goal is to find data-driven techniques capable of modeling specific simulation outcomes, such as mode choice, distance traveled, trip duration, and waiting time. The Sioux Falls scenario was used as an example to implement the suggested solution since it has one of the simpler road networks compared to the other scenarios. The scenario's goal is to deliver a realistic, completely dynamic demand with a diverse socio-demographic user base and a high level of geographical detail (Chakirov and Fourie., 2014). All of the information used in the simulation scenario, such as networks, facilities, locations, land use, building information, and census data, reflect the main characteristics of Sioux Falls; however, the scenario is not intended to replicate the actual city and remains a fictitious test scenario. One of the key elements of the approach proposed is that we use data generated with an agent based simulation. Therefore, we do not only have perfect and complete observations, but we also know the models that are producing such observations.

In the first stage, data is acquired from the simulation's outcome. The simulation platform MATSim was used, an agent-based model largely applied in the academic field and here used to produce synthetic data. In MATSim, individual agents seek to optimize their daily activity plans, and this is done in an iterative manner, until equilibrium is reached (Horni et al., 2016). Many parameters exist in MATSim to define the behavior of the agents, the transportation system, and the behavior of the simulation itself. Some of them are used here to define and implement the problem's conditions. The configuration file is the source that contains all the parameter types and defines some of the properties of the scenario.

Data preparation will then handle the simulation results produced from the Sioux Falls scenario. After merging all of the results into different databases, a number of processes, including data cleaning and preprocessing, have been carried out in preparation for machine learning modeling.

A set of machine learning techniques was applied to model different variables such as mode of transportation, distance traveled, travel time, and waiting time, which reflect the decision making of the traveler. First of all, these variables will be modeled on the first database (made from default simulation parameters) using the most efficient techniques. Furthermore, once built, the model's parameters must be tuned in order to ensure the highest efficiency. These trained models will be evaluated and saved for further testing. The model's performance must be validated using a variety of metrics. In the case of classification problems, precision, in addition to error of classification, will be the most important indicators to consider when evaluating the model on new data. In classifying data, it is critical to pay attention to the ROC-AUC curve since it reveal how effectively the model selected the correct class. The confusion matrix, which contains all of the combinations of the correct and incorrect categories, will then be reviewed to help us determine which of the classes is difficult to anticipate. Other measures, like F1 score and recall, will indicate the model's strength. Furthermore, R-squared and the Root Mean Square Error (RMSE) are the most commonly used and relevant metrics to consider while dealing with regression problems.

Finally, the objective of the study requires the development of numerous scenarios to cover as many real-world situations as possible and to examine the robustness of trained machine learning models to new situations. To do

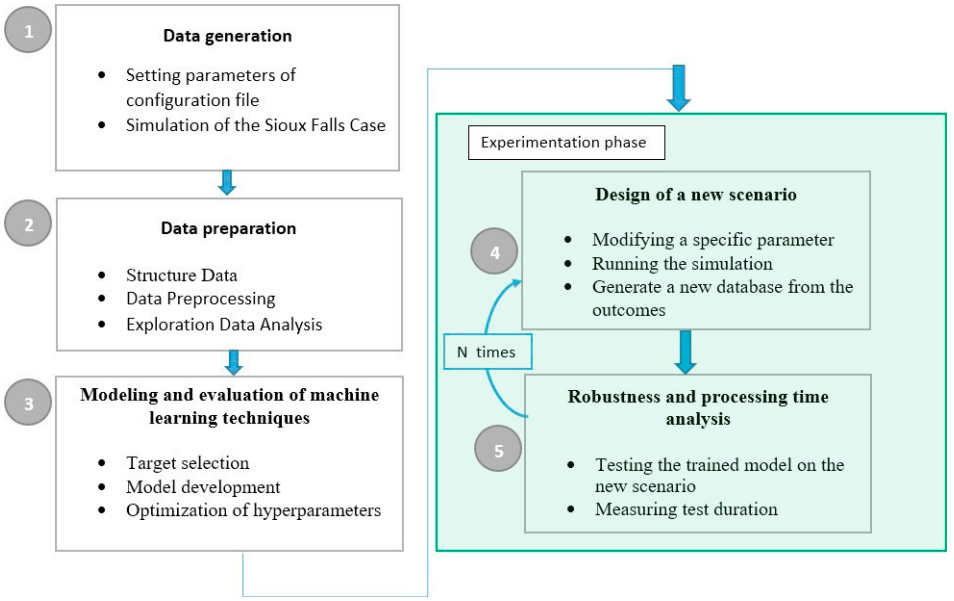

Fig. 1. Proposed approach

this, a single parameter of simulation will be changed in each experiment, and then the simulation will be performed to create new data that represent a different situation. Then, the models that were trained using the initial database and default parameters will be tested these new situations to determine their robustness. Three distinct models were examined in order to predict four outcomes: trip distance, travel duration, mode choice, and waiting time. Because of the variety of models, multiple forms of preprocessing are necessary to attain the best performance. The goal of this part is to see if machine learning techniques can recognize change in different circumstances and how robust they are to parameter variability. The prediction time of the techniques will be measured in seconds and examined to determine which models will require the least amount of testing time. All these parts represent the methodology outlined in five sections and illustrated in Fig. 1.

4. Scenario and data

4.1. Simulation Scenario

The Sioux Falls scenario simulation environment will be used to perform the necessary experiments because it has one of the simplest road networks. Scenario is a common term in transportation that the case architecture, containing the network, facilities, transportation modes, agent plans, and so on, as well as specific parameters. The planning of scenario according to Lyons et al. (2021) is a process that analyzes the impact of various policies, plans, and/or programs on the future of a community or region. This activity can help decision-makers develop transportation plans by providing information.

The first step is to adjust the configuration file, which contains all of the parameters and inputs required to generate the data and begin the simulation. The scenario's architecture will be built by entering the network, which contains a description of all the nodes and links. Each node is distinguished by specific coordinates, and each link connects two nodes and has a specific capacity, free speed, and length. in addition to programming all transit route stops based on the movement of the bus (each bus has an arrival and departure schedule). All types of facilities, modes of transport and agents have specific characteristics that ensure and describe their uniqueness.

The population file will represent the various actions proposed by the agents; each agent has a memory that allows him to remember a certain number of plans, and each plan includes all of the trips made in a day. Aside from the input

Table 1. Definition of parameter variation in the experiments

Parameter	Description	Original	New
P1	–	–	–
P2	monetaryDistanceRate (car)	-0.0004	-0.4
P3	monetaryDistanceRate (car)	-0.0004	-0.004
P4	marginalUtilityOfTraveling-util-hr (pt)	-0.18	-0.0018
P5	marginalUtilityOfTraveling-util-hr (walk)	-1.14	-100.14
P6	waitingPt	-0.18	-100.18
P7	maxBeelineWalkConnectionDistance	300	0
P8	waitingPt	-0.18	1
P9	lateArrival	0	-100
P10	marginalUtilityOfMoney	0.062	100
P11	earlyDeparture	0	0.005
P12	marginalUtilityOfTraveling-util-hr (car)	0	-0.005
P13	marginalUtilityOfDistance-util-m (walk)	0	-0.1
P14	marginalUtilityOfMoney	0.062	0.1
P15	utilityOfLineSwitch	0	0.3
P16	marginalUtilityOfTraveling-util-hr (pt)	-0.18	-0.5
P17	monetaryDistanceRate (pt)	0	-0.1
P18	monetaryDistanceRate (pt)	0	-0.0003
P19	learning rate	1	0.8
P20	search Radius	1500	2500

files, there are numerous parameters related to the plan score, strategies, and so on. The simulation was run over 1000 iterations to achieve equilibrium, which takes at least 8 hours.

4.2. Experiments description

To produce varied scenarios, several parameters in MATSim's configuration file were changed, and for each new parameter, the simulation was affected, followed by the creation of a new database based on the new outcomes.

The aim of parameter variation is to obtain different situations from that created by the default parameters. So, for each scenario, we choose one parameter to modify and try to vary its values and see its effect on the outcome, i.e. to see if the data resulting from the new scenario are different from those generated by the default scenario. If the data doesn't change much based on exploratory data analysis methods, we modify the value again to obtain a higher relative change between the default and the new value, until we obtain data (information) that is different from that generated by the default scenario. Some parameters require a high relative change to observe their effect on the data obtained, while others require a lower relative change. Some parameters have a default value of zero; we changed them to different values, strictly positive or negative, with a low relative change in most cases, which is sufficient to see its effect on the new data. The objective of obtaining data different from the original data is to create a database enabling us to test the robustness of the trained models that will be examined later in this paper. Following that, the model trained on the first scenario (the one generated from default parameters) will be used to test the newly created situations, which will be evaluated on the basis of specific metrics like accuracy and root mean squared error in the next part; all of these experiments that represent parameter changes are shown in Table 1, and it should be noted that the first experiment is performed on the original configuration file (default parameters).

By adjusting the simulation parameters and establishing different circumstances, new databases with distinct patterns will be produced. Since these are the significant variables in simulation, the models of waiting time, distance traveled, travel duration, and mode selection will be evaluated to assess their strength.

4.3. Data preparation

This part will describe the steps involved in building the database that will serve as the preparation for the modelling. Data about agents (people), output legs, generated trips, output of transit vehicles, different networks, and output facilities are just a few of the related pieces of information for the simulation's output file. These results were all prepared and transformed by data cleaning and pre-processing techniques to get a final database. Other features were also developed to reinforce the database and provide a clear view of other specific issue.

Table 2. Description of the database features.

Feature	Description	Type
person	The ID of an agent	Categorical
executed score	The final score for an agent	Float
sex	Gender of the agent	Bool
age	Age of the agent	Integer
carAvail	The availability of the car	Bool
employed	Employment status	Bool
trip number	The number of the current trip in the plan	Integer
traveled distance	The duration of all trips	Float
euclidean distance	The Euclidean distance of the path	Float
longest distance mode	The mode with the greatest distance	Categorical
modes	The different types of modes in a plan	Categorical
start activity type	The original activity of an agent	Categorical
end activity type	The last activity of an agent	Categorical
dep time	Time of departure	Datetime
trav time	The duration of the trip	Float
wait time	The waiting time	Float
distance	The distance of a trip	Float
mode transport	The mode chosen in a trip	Categorical
start link	The first route or connection of the trip	Categorical
end link	The last route or connection of the trip	Categorical
$start_x$	The first coordinate of the starting location	Float
$start_y$	The second coordinate of the starting location	Float
end_x	The first coordinate of the final location	Float
end_y	The second coordinate of the final location	Float
transit route	The transit route	Categorical

At the end of the procedure, a large number of variables were produced that could be classified based on their standards; all of the information defining the final database are provided in Table 2. Following the collection of simulation results and leading up to the final database, data cleaning is an important step, it entails ensuring that the data is accurate, consistent, and usable (Ridzuan et al., 2019). As a result, it identify errors or corruption, correcting or removing them, and, if necessary, manually processing the data to prevent the same errors from occurring again. The final database's features could be divided into four categories: data about personal characteristics, trips, plans, and network. The data has been cleaned and the main transformations are presented in the following format:

- Merge data sets
- Delete unwanted data
- Process the Data-time variables
- Normalise Data

- Verify the results
- Export the data

Following the completion of data cleaning and missing value processing, additional part occurs in the process by transforming the required output to the appropriate type for the model; some methods have been used here, beginning with Min-Max normalisation, which subtracts the minimum value in the feature and then divides by its range. The difference between the original maximum and minimum is the range. Finally, it reduces all values to a fixed range of 0 to 1. Depending on the range of the classes and the role of the feature in the model, two types of methods were used for categorical variables. The first method is label encoding, which simply converts each categorical value in a column to a number. To convert to binary variables, label encoding is recommended. The second method is one-shot encoding, which is by far the most common way to represent categorical variables. This method is also known as dummy variables. Dummy variables are used to replace a categorical variable with one or more new features that can have values between 0 and 1.

5. Experimentation results

After completing data preparation, all is ready for the experimental stage, but only the preprocessing part can be modified based on the type of model. First, tests will be performed to classify the various modes of transportation and predict travel time, waiting time, and distance travelled using multiple types of models chosen based on an efficient algorithm. Based on their relevance in the simulation and decision-making process, the previously indicated features were chosen from among all feasible variables. Furthermore, using the available data, the Random Forest approach was employed to estimate the choice of passenger transportation mode and by applying the feature importance technique, it seems that this output is influenced by three significant factors that typically describe the needs for a given outcome. According to the results of this test, the most important factors are distance traveled, travel time and waiting time. The same feature importance test was then performed on the models of these variables to reveal that the common factors including mode choice, distance and duration of travel, and waiting time are the ones that drive the proposed data-driven techniques.

5.1. Model selection

An algorithm for model selection was used to evaluate the possible techniques for each output. The performance of the model will be measured by the accuracy and the root mean squared error of training and testing in each experiment.

The results show that the random forest is the most effective technique for all outputs due to its well-structured process, but it consumes time while testing, taking an average of 20 minutes for regression problems and less than 5 minutes for classification. XGBoost, on the other hand, has excellent results in terms of accuracy and error, with a low average consuming time of 0.65 minutes.

The artificial neural network is also one of the models with a good reputation in the field of machine learning for regression and classification problems that will be encountered in the following experiments and compared to the results of other models. The selected data-driven techniques are:

- Artificial neural networks
- Random forest
- Extreme Gradient Boosting

5.2. Evaluation of decision variable models

The first variable to be modeled is the choice of transport mode, it is a classification problem and two techniques will be applied in this direction. Artificial neural networks (ANN) has been trained using cross validation on 10 epochs, a batch size of 100, and categorical cross entropy loss. This model's result shows an accuracy of 99.8% and a validation loss of 0.019, indicating a powerful model. Relu is the activation function for the hidden layers, and softmax is the activation function for the output layer.

The second technique is the random forest, which shows a great ability to understand the patterns of the data. By applying this model on the existing dataset; it gives 99.9% accuracy and RMSE equal to 0.018 in the training set. For the test part, we obtained 98.6% accuracy and RMSE of 0.019, which are both good results. For parameter optimization, a grid search algorithm, detailed in Bergstra and Bengio (2012), was used to find the best combination of parameters from a wide range of values for each attribute. Table 3 will present the results for the remaining outputs

Table 3. Results of variables modeling.

Experiment	Artificial neural networks				Random forest				XGBoost			
Output	Train acc	Test acc	Train RMSE	Test RMSE	Train acc	Test acc	Train RMSE	Test RMSE	Train acc	Test acc	Train RMSE	Test RMSE
Traveled distance	96.6	93.3	0.027	0.032	99	93.2	0.018	0.034	96	94.5	0.023	0.029
Trip duration	91.8	91.2	0.012	0.013	95.5	90.6	0.011	0.0148	96.6	92.2	0.011	0.012
Waiting time	90.3	88	0.035	0.039	97.1	89.9	0.032	0.035	90.8	88.7	0.034	0.036

which are travelled distance, trip duration, and waiting time. The models used here will deal with a regression problem; the goal is to demonstrate the existence of data-driven techniques that can understand the patterns behind the data. Three models have been proposed to solve these tasks; for the travelled distance, the results are so close with a small advantage for the random forest model, and for trip duration, the same is true for XGBoost. When it comes to waiting time, the random forest regressor outperforms other models with an accuracy of 97.1%. To achieve maximum efficiency, all models included a parameters optimization process.

5.3. Model robustness analysis for mode choice prediction

The table 4 focuses on the model's ability to predict the mode choice variable in each experiment. In general, the test accuracy is above 90% in most cases, which validate the effectiveness of the training and allows us to state that the two trained models, ANN and Random Forest, can recognize the built-in models in MATSim.

Although both trained models perform equally well, the random forest is more powerful at predicting new data, with an accuracy that is always greater than 99%, making this technique more robust than the artificial neural network. The input data is processed using MinMaxscaler method to vary the values between 0 and 1. This step saves training processing time and reduces computational work. The model will perform worse after various experiments if other techniques are used. In terms of test time, the random forest does not exceed 3 seconds. Because of the complexity of the links between the neurons, the ANN has a shorter test time, on the order of 10 seconds, but it is still longer than the case of random forest.

P2 has a high variability of the car monetary distance rate ranging from -0.0004 to -0.4, however the ANN has an accuracy of 87%. By decreasing this variability to only 10% in P3, the model recognizes the data better, with an accuracy of 94%. As a result, the variability of this parameter has a significant impact on the mode selection outcome. On the other hand, there was a different type of change in the monetary distance rate for public transportation, from 0 to -0.1 in P17 and to -0.0003 in P18, both of which are very similar in terms of their ability in prediction. The variability of this parameter has small impact on the output. Despite the large variability in P9, late arrival has little effect on mode choice and the 0.005 reward for an early departure does not change the traveler's decision. P10 and P14 describe the change in the marginal utility of money, according to the ANN model, changing this value from 0 to 100 in the first stage and to 0.1 in the second stage decreases the accuracy by 3% and 9%, respectively, but the model still has a good ability to recognize this change. Despite the high gap of change in P16 and P4, the proposed models could recognize the new patterns exists in the data even with high variation. In the case of high performance, it is difficult to determine whether the model is so effective at comprehending new information or the parameter has no dependence on the output. Because it is based on specific weights from the training, the ANN could reveal this reality of this point. The majority of the other experiments show that the trained models can predict new scenarios, leading us to conclude that these techniques are able to replicate mode choice models in MATSim.

5.4. Model robustness analysis for trip distance prediction

The distance traveled as an output is one of the main elements that drive the simulation by taking part in the scoring function, furthermore, it is considered one of the important features of the mode choice model, which makes

Table 4. Results of mode choice prediction.

Model	Artificial Neural Networks		Random Forest	
Metric	Accuracy (%)	Time (s)	Accuracy (%)	Time (s)
P1	99	9.13	99.94	1.4
P2	87	14.09	99.92	2.6
P3	94	13.6	99.93	2.7
P4	97	9.04	99.94	1.3
P5	92	8.53	99.3	1.2
P6	94	5.56	99.64	0.8
P7	96	7.75	99.93	1.1
P8	98	9.75	99.89	1.5
P9	96	8.93	99.86	1.3
P10	87	15	99.7	2.6
P11	96	10.4	99.92	1.5
P12	94	11.4	99.88	1.4
P13	91	8.81	99.85	1.3
P14	90	8.7	99.82	1.6
P15	91	9.7	99.86	1.3
P16	96	8.76	99.93	1.3
P17	96	8.9	99.93	1.2
P18	97	5.16	99.94	0.8
P19	95	9.26	99.93	1.3
P20	97	9.96	99.94	1.3

it beneficial to model this type of variable and study its ability to reproduce the pattern responsible for the creation of the distance traveled data. A new machine learning technique appears in this regard, XGBoost, which has a greater ability than ANN and random forest to reduce the time spent to test other scenarios, the prediction will occur in less than half a second with the same performance as the random forest model. Despite the good results of ANN, it does not achieve the level of robustness demonstrated by other models. This is due to the complex architecture of the trained technique, which is based on sets of weights with specific values, whereas the random forest makes decisions based on a range of values, giving the model greater capacity to identify the right action.

P6, which represents a substantial change in the waiting time parameter, has a measurable influence on the trained model's predictions, so the accuracy of the ANN decreasing from 97% to 70% and it also decreases in the random forest and XGBoost. The waiting parameter influences the distance traveled by altering the passenger's strategies, which may cause him to travel a longer distance. In P8, the agent will try to find the best transit route to maximize the plan score; in this case, the change is in the opposite direction, implying that more waiting time is beneficial. In terms of model results, we conclude that data-driven techniques can estimate the recommended output regardless of variation. This means that the distance traveled and the delay waiting have a high correlation. P2 and P3 demonstrate the same statements made in the mode choice prediction where the distance traveled is so sensitive to these variations. Changing the learning rate in P19 from 1 to 0.8 has little effect on the performance of trained models in terms of accuracy and test time. The increase in search radius, on the other hand, has a negative impact because it allows the human to look for public transportation much farther, resulting in an increase in walking distance. In the random forest and ANN, the accuracy in P20 declines from its value in the original scenario, but the XGBoost was able to recognize this variation where the accuracy remained constant. By increasing the marginal utility of money from 0.062 to 100, the number of car users falls dramatically, and the substitute is public transportation, which lengthens the path and makes it more difficult for the trained models to acknowledge the new environment, explaining the drop in accuracy in P10 experiment. If this parameter is changed to a high negative value, there will be no substantial shift in the data due to the defined conditions, for example, travelers who do not own a car will always use public transportation,

Table 5. Results of traveled distance prediction.

Model	Artificial Neural Networks		Random Forest		XGBoost	
Metric	Accuracy	Time (s)	Accuracy	Time (s)	Accuracy	Time (s)
P1	97	11.9	99	1.73	96	0.26
P2	70	17.99	96	2.55	93	0.35
P3	85	17.8	98	2.40	96	0.36
P4	95	11.77	97	1.77	95	0.23
P5	85	10.93	95	1.56	93	0.21
P6	70	6.80	91	0.90	89	0.09
P7	92	9.8	98	1.50	95	0.20
P8	80	12.62	98	1.85	96	0.25
P9	90	10.95	95	1.60	94	0.24
P10	74	18.8	96	2.4	92	0.36
P11	87	11.2	93	1.73	94	0.25
P12	86	11.48	91	1.94	93	0.29
P13	84	11.05	89	1.61	92	0.23
P14	80	11.63	85	1.77	89	0.24
P15	85	11.59	91	1.52	93	0.21
P16	91	11.21	93	1.65	95	0.23
P17	92	11.18	94	1.73	95	0.25
P18	93	7.14	95	0.99	96	0.14
P19	95	11.23	93	1.60	95	0.22
P20	88	11.5	94	1.62	96	0.24

regardless of the marginal utility of the money. Looking at all of the table 5, it is clear that each model recognizes a specific scenario more than the other models, indicating that no one technique dominates in predicting the variation that affects distance travelled. This analysis also reveals that these techniques are less effective at predicting travel distance than mode choice, and the correlation of these parameters with travel distance is higher than that of mode choice.

5.5. Model robustness analysis for trip duration prediction

The third variable that must be checked using the same models as in the previous part is travel time. The results show that the trained architecture was able to recognize the majority of the parameter variations; however, the XG-Boost still has the best result in terms of test time, and in this case, it also has the best performance in understanding all of the new scenarios based on the accuracy in comparison to the previous experiment related to travel distance. Despite its high scores, the ANN model still struggles to recognize new data based on specific scenarios such as P3, P5, P8, P13 and P6. Aside from P5, random forest and XGBoost are very good at predicting travel time in the other scenarios. P5 represents the change in marginal travel utility per hour for walking, with a large variability of the parameter ranging from -1.14 to -100.14, resulting in a considerable variation in travel time. Walking will be extremely difficult in this situation, and only car owners will be unaffected. This case makes it difficult for models to predict this output due to the high correlation between this parameter and output. P13 represents the change in marginal utility of distance (by walking) and presents the same difficulty as P5, where the accuracy is 63% when looking at the ANN model. Changing the late arrival value from 0 to -100 causes significant variation in the trip duration data, making it challenging for the trained models to identify new trends. In P6, increasing the waiting time penalty reduces the number of walkers and public transportation users; in P8, the opposite occurs. As shown in the table, both of these changes are difficult for ANN to detect, so the trained XGBoost model with the best performance will be recommended.

Table 6. Results of trip duration prediction.

Model	Artificial Neural Networks		Random Forest		XGBoost	
Metric	Accuracy	Time (s)	Accuracy	Time (s)	Accuracy	Time (s)
P1	91	11.3	96	1.6	97	0.28
P2	74	18.24	96	2.6	97	0.49
P3	62	17.6	95	2.5	96	0.44
P4	88	11.7	90	1.6	92	0.3
P5	35	10.4	60	1.4	71	0.26
P6	51	6.7	93	0.9	94	0.17
P7	89	9.8	95	1.5	96	0.24
P8	65	12.6	84	1.8	86	0.32
P9	72	11.4	73	1.6	72	0.29
P10	66	18.59	96	2.9	97	0.46
P11	80	12.6	89	2.02	91	0.28
P12	79	12.5	85	2.04	88	0.25
P13	63	11	74	1.52	78	0.25
P14	73	11.6	79	1.2	80	0.23
P15	79	10.4	87	1.49	90	0.25
P16	82	11.1	90	1.47	92	0.27
P17	83	10.9	90	1.48	92	0.26
P18	85	6.85	93	0.95	94	0.15
P19	90	11.2	89	1.5	91	0.28
P20	78	11.24	90	1.48	92	0.26

Other experiments show that the three models are robust, with an accuracy of more than 70% in general. Despite some weaknesses in the ANNs, the rest of the experiments confirm the trained models' high robustness. XGBoost is the recommended technique for the trip duration modeling , not only for its ability to understand new scenarios, but also for its low testing time.

5.6. Model robustness analysis for waiting time prediction

This part will examine the strength of models in order to predict the final significant variable that drives the simulation, which is the waiting time; this output is correlated with the mode choice feature by assisting the traveler in determining the route that guarantees the least amount of waiting time. The trained models were tested in a variety of scenarios to determine their robustness, which will be assessed using R-squared. The test time for each test will also be examined in order to determine the best match between robustness and test duration. The random forest outperforms the other accuracy-based techniques in this case; however, XGBoost is the best solution for the test time.

The most influential experiment on waiting time prediction is P6, which has a high variability. Despite the fact that the results are not as good as the previous experiments, the model still performs well based on XGBoost and random forest trained models with an accuracy greater than 80%. When the number of car users decreases, it appears that P2 and P3 have a considerable impact on the distribution of waiting time in the data, because most travelers choose to use public transportation. In this case, the ANN model struggles to grasp the new information based on accuracy. The remaining experiments demonstrate that the suggested technique is capable of reproducing the patterns responsible for generating the waiting time variable. The purpose of the previous tables was to analyze the robustness of modeling mode choice, distance traveled, travel time, and waiting time across various experiments, in addition to analyzing the test time of each ML technique.

In terms of robustness, Random Forest is the best model for predicting mode choice, travel distance and waiting time, while XGBoost is the best for predicting travel time. In terms of test time, the XGBoost model can provide results

Table 7. Results of waiting time prediction.

Model	Artificial Neural Networks		Random Forest		XGBoost	
Metric	Accuracy	Time (s)	Accuracy	Time (s)	Accuracy	Time (s)
P1	89	20.5	98	8	91	0.24
P2	65	18.3	89	7.6	85	0.37
P3	69	17.4	93	8.5	90	0.36
P4	88	12.04	91	8.1	88	0.25
P5	77	11.1	84	7.5	82	0.23
P6	64	7.1	83	4.4	81	0.15
P7	80	10.2	89	6.7	87	0.23
P8	73	12.7	91	7.6	89	0.26
P9	82	10.94	86	6.8	85	0.25
P10	69	18.27	89	7.1	85	0.36
P11	79	13.3	90	5.05	91	0.27
P12	78	12.8	87	5.6	88	0.25
P13	68	11.43	84	5.21	82	0.21
P14	70	10.9	85	5.6	83	0.29
P15	76	10.89	87	6.7	86	0.26
P16	83	11.81	90	5.08	87	0.21
P17	84	13.01	90	5.09	89	0.21
P18	85	7.95	92	2.57	90	0.14
P19	87	11.34	91	4.98	88	0.22
P20	77	12.6	90	5.02	88	0.23

in less than one second. Random forest performs less well in this regard, with an average test time of 6 seconds, and artificial neural networks have an average testing duration of 11 seconds. This leads us to conclude the existence of data-driven techniques that perform well enough and can reproduce the models behind the generation of these variables in the simulation.

6. Conclusion

To face new transportation and traffic challenges, it is important to enhance the quality models that inform policy making. Agent-based modeling is a powerful method for assessing hypothetical scenarios before they are implemented in reality. However, computation time is one of its main limitations and, according to several researchers, machine learning could help overcome it. The first step in order to obtain such result is to prove that some of the behavioral models integrated that are typically embedded in agent-based simulations, can be approximated by machine learning models. This was done taking MATSim, a popular agent-based simulation, and using machine learning to reproduce some of its output. An exploratory analysis of the data was performed in order to understand how output varies for different scenarios. Next, on this artificial data, XGBoost, artificial neural networks, and random forest, three complex machine learning models, were applied. They performed well in understanding the patterns of artificial data generated using the simulation's default parameter values. All of the data-driven techniques had an accuracy of greater than 90% across all experiments for predicting travel distance, travel time, and waiting time. Further tests showed the ability of trained machine learning models to recognize other scenarios, demonstrating the robustness of these techniques. For predicting all features, the results were best for random forest and XGBoost, and less successful for artificial neural networks. In all cases, XGBoost had the low processing time to less than one second and the best robustness in predicting travel time. Random Forest, on the other hand, predicts travel distance, waiting time, and mode choice with high precision and in less than 10 seconds, proving its ability to reproduce agent-based models' output related

to these variables. The research might be expanded by investigating the application of data-driven techniques in the agent-based model process or by developing some kind of combination of them.

References

Angione, C., Silverman, E., Yaneske, E. (2022). Using machine learning as a surrogate model for agent-based simulations. PLoS ONE, 17(2).

Antoniou, C., Koutsopoulos, H.N., (2006). Estimation of traffic dynamics models with machine learning methods. Transp. Res. Rec. J. Transp. Res. Board 1965, 103–111.

Antoniou, C., Koutsopoulos, H.N., Yannis, G., (2013). Dynamic data-driven local traffic state estimation and prediction. Transp. Res. C Emerg. Technol. 34, 89–107.

Azlan, N.N.N., Rohani, M.M., (2018). Overview of application of traffic simulation model. MATEC Web Conf, 150, 03006.

Bálint, K., Tamás, T., Tamás, B., (2022). Deep Reinforcement Learning based approach for Traffic Signal Control. Transportation Research Procedia, 62, 278–285.

Bergstra, J., Bengio, Y. (2012). Random Search for Hyper-Parameter Optimization. Journal of Machine Learning Research, 13(2012), 281–305.

Bhavsar, P., Safro, I., Bouaynaya, N., Polikar, R., Dera, D., (2017). MACHINE LEARNING IN TRANSPORTATION DATA ANALYTICS. Data Analytics for Intelligent Transportation Systems, 283–307.

Brearcliffe, D.K., Crooks, A., (2021). Creating Intelligent Agents: Combining Agent-Based Modeling with Machine Learning. Conference of The Computational Social Science Society of the Americas.

Cao, K. (2022). A Machine Learning-Based Approach to Railway Logistics Transport Path Optimization. Mathematical Problems in Engineering, 2022.

Chakirov, A. Fourie, P., (2014). Enriched Sioux Falls Scenario with Dynamic and Disaggregate Demand, Working paper, Future Cities Laboratory, Singapore - ETH Centre (SEC), Singapore.

Chen, T.‘ Guestrin, C., (2016). XGBoost: A scalable tree boosting system. Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, New York, NY, USA, pp. 785–794.

Chu, K.F., Lam, A.Y.S., Loo, B.P.Y., Li, V.O.K., (2019). Public Transport Waiting Time Estimation Using Semi-Supervised Graph Convolutional Networks. IEEE Intelligent Transportation Systems Conference, ITSC 2019. Institute of Electrical and Electronics Engineers Inc., pp. 2259–2264.

De Souza, F., Verbas, O., Auld, J., (2019). Mesoscopic traffic flow model for agent-based simulation. Procedia Computer Science. Elsevier, pp. 858–863.

Dong, Y., Sun, Y., Waygood, O., Wang, B., Huang, P., Naseri, H., (2022). Insight into the Nonlinear Effect of COVID-19 on Well-Being in China: Commuting, a Vital Ingredient. J. Transp. Heal.

Frick, R. (2011). Simulation of transportation networks. Conference: Proceedings of the 2011 Summer Computer Simulation Conference, 188–193.

Furtado, B.A., Andreao, G.A. (2022). Machine Learning Simulates Agent-Based Model Towards Policy.

Hong, Z., Son, Y.J., Chiu, Y.C., Head, L., Feng, Y., Xi, H., Kim, S., Hickman, M., (2013). A Primer For Agent-Based Simulation And Modeling In Transportation Applications. U.S. Department of Transportation. Federal Highway Authority (FHWA), 201.

Horni, A., Nagel, K. and Axhausen, K.W., Eds. (2016). The Multi-Agent Transport Simulation MATSim. Ubiquity Press, London.

Huang, J., Cui, Y., Zhang, L., Tong, W., Shi, Y., Liu, Z., (2022). An Overview of Agent-Based Models for Transport Simulation and Analysis. Journal of Advanced Transportation.

Huang, W., Song, G., Hong, H., Xie, K., (2014). Deep architecture for traffic flow prediction: deep belief networks with multitask learning. IEEE Trans. Intell. Transp. Syst. 15 (5), 2191-2201.

Hung, C.Y., Chen, W.C., Lai, P.T., Lin, C.H., Lee, C.C., (2017). Comparing deep neural network and other machine learning algorithms for stroke prediction in a large-scale population-based electronic medical claims database. Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS.

Hunter, E., Kelleher, J.D., (2022). Validating and Testing an Agent-Based Model for the Spread of COVID-19 in Ireland. Algorithms 15, no. 8: 270.

Jamal, A., Zahid, M., Tauhidur Rahman, M., Al-Ahmadi, H.M., Almoshaogeh, M., Farooq, D., Ahmad, M., (2021). Injury severity prediction of traffic crashes with ensemble machine learning techniques: a comparative study. Int. J. Inj. Contr. Saf. Promot.

Jenelius, E., Koutsopoulos, H.N., (2013). Travel time estimation for urban road networks using low frequency probe vehicle data. Transp. Res. B Methodol. 53, 64-81.

Jeon, H., Seo, W., Park, E., Choi, S., (2020). Hybrid machine learning approach for popularity prediction of newly released contents of online video streaming services. Technol. Forecast. Soc. Change 161, 120303.

Kagho, G.O., Balac, M., Axhausen, K.W., (2020). Agent-based models in transport planning: current state, issues, expectations. Procedia Comput. Sci. Elsevier B.V. 170, 726—732.

Kayadelen, C., Önal, Y., Altay, G., Öztürk, M., Serin, S., (2022). Effects of maintenance, traffic and climate condition on International Roughness Index of flexible pavement. Int. J. Pavement Eng.

Llorca, C., Kuehnel, N., Moeckel, R., (2020). Agent-based integrated land use/transport models: A study on scale factors and transport model simulation intervals. Procedia Computer Science. Elsevier, pp. 733–738.

Lv, Y., Duan, Y., Kang, W., Li, Z., Wang, F.Y., (2015). Traffic flow prediction with big data: a deep learning approach. IEEE Trans. Intell. Transp. Syst. 16 (2), 865-873.
Lyons, G., Rohr, C., Smith, A., Rothnie, A., Curry, A., (2021). Scenario planning for transport practitioners. Transportation Research Interdisciplinary Perspectives, 11.
Maniruzzaman, M., Rahman, M.J., Al-MehediHasan, M., Suri, H.S., Abedin, M.M., El-Baz, A., Suri, J.S., (2018). Accurate Diabetes Risk Stratification Using Machine Learning: Role of Missing Value and Outliers. J. Med. Syst. 42, 1–17.
Naseri, H., Jahanbakhsh, H., Hosseini, P., Moghadas Nejad, F., (2020). annsy developing a new machine learning technique. J. Clean. Prod. 258, 120578.
Naseri, H., Waygood, E.O.D., Wang, B., Patterson, Z., Daziano, R.A., (2021). A Novel Feature Selection Technique to Better Predict Climate Change Stage of Change. Sustainability 14, 40.
Naseri, H., Jahanbakhsh, H., Foomajd, A., Galustanian, N., Karimi, M.M., Waygood, E.O.D., (2022). A newly developed hybrid method on pavement maintenance and rehabilitation optimization applying Whale Optimization Algorithm and random forest regression.
Naseri, H., Waygood, E.O.D., Wang, B., Patterson, Z., (2022). Application of Machine Learning to Child Mode Choice with a Novel Technique to Optimize Hyperparameters. Int. J. Environ. Res. Public Health 19, 16844.
Naseri, H., Waygood, E.O.D., Wang, B., Patterson, Z., (2023). Interpretable Machine Learning Approach to Predicting Electric Vehicle Buying Decisions. Transp. Res. Rec. J. Transp. Res. Board 036119812311695.
Nguyen-Sy, T., Wakim, J., To, Q.D., Vu, M.N., Nguyen, T.D., Nguyen, T.T., (2023). Predicting the compressive strength of concrete from its compositions and age using the extreme gradient boosting method. Constr. Build. Mater. 260, 119757.
Pell, A., Meingast, A., Schauer, O., (2017). Trends in Real-time Traffic Simulation. World Conference on Transport Research - WCTR 2016 Shanghai. 10-15 July 2016, 1477—1484.
Poongodi, M., Malviya, M., Kumar, C., Hamdi, M., Vijayakumar, V., Nebhen, J., Alyamani, H., (2022). New York City taxi trip duration prediction using MLP and XGBoost. Int. J. Syst. Assur. Eng. Manag. 13, 16–27.
Ridzuan, F., Nazmee, W.M., Zainon, W., (2019). A Review on Data Cleansing Methods for Big Data. Procedia Computer Science 161 (2019), 731—738.
Santos, F., Nunes, I., Bazzan, A.L.C., (2020). Quantitatively Assessing the Benefits of Model-driven Development in Agent-based Modeling and Simulation. Simulation Modelling Practice and Theory.
Sivakumar, N., Mura, C., Peirce, S.M., (2022). Combining Machine Learning and Agent-Based Modeling to Study Biomedical Systems. Machine Learning Agent-based Modeling of Biomedical Systems.
Sun, P., Aljeri, N., Boukerche, A., (2020). Machine Learning-Based Models for Real-time Traffic Flow Prediction in Vehicular Networks. IEEE Network, vol. 34, no. 3, 178–185.
Sun, Y., Dong, Y., D. Waygood, E.O., Naseri, H., Jiang, Y., Chen, Y., (2022). Machine-Learning Approaches to Identify Travel Modes Using Smartphone-Assisted Survey and Map Application Programming Interface. Transp. Res. Rec. J. Transp. Res. Board 036119812211064.
Tizghadam, A., Khazaei, H., Moghaddam, M.H.Y., Hassan, Y., (2019). Machine Learning in Transportation. Journal of Advanced Transportation, 2019 (2019) 3 pages.
Wang, Y., Sherry Ni, X., (2019). A XGBOOST risk model via feature selection and bayesian hyper-parameter optimization. Int. J. Database Manag. Syst. 11, 01–17.
Zhang, W., Valencia, A., Chang, N.B., (2021). Synergistic Integration Between Machine Learning and Agent-Based Modeling: A Multidisciplinary Review. IEEE Trans Neural Netw Learn Syst.
Zheng, G., Liu, H., Xu, K., and Li, Z., (2021). Objective-aware Traffic Simulation via Inverse Reinforcement Learning. International Joint Conference on Artificial Intelligence 2021.
Zheng, J., Suzuki, K., Fujita, M., (2013). Car-following behavior with instantaneous driver-vehicle reaction delay: a neural-network-based methodology. Transp. Res. C Emerg. Technol. 36, 339-351.