IBIMA Publishing
Communications of International Proceedings
https://ibimapublishing.com/p-articles/46AI/2025/4637625/
Vol. 2025 (34), Article ID 4637625

Machine Learning to Support the Effectiveness of Physical Training*

Bartosz OCIMEK and Joanna WIŚNIEWSKA

Institute of Information Systems, Faculty of Cybernetics,
Military University of Technology, Gen. S. Kaliskiego 2, 00-908 Warsaw, Poland

Correspondence should be addressed to: Bartosz OCIMEK, artosz.ocimek@gmail

* Presented at the 46th IBIMA International Conference, 26-27 November 2025, Ronda, Spain

Abstract

This study investigates the use of machine learning methods to assess and predict the level of athletes' preparation for e-cycling competitions based on their training data. A three-month dataset of activities collected from selected competitors of the Polish Open E-cycling Championships was processed to extract key performance indicators describing individual training cycles. Three supervised learning models – linear regression, decision trees, and artificial neural networks – were evaluated, achieving average classification accuracy of approximately 0.6. Additionally, an unsupervised k-means clustering approach was applied to identify natural groupings of athletes based on multidimensional training characteristics. The findings indicate that the constructed dataset enables reasonable prediction of preparation level; however, broader and more diverse data, including contextual factors such as well-being, nutrition, and environmental conditions, may be required to significantly improve model performance. The results highlight both the potential and the limitations of applying machine learning techniques to support planning and evaluation of training in e-cycling.

Keywords: machine learning, e-sport, artificial intelligence, data science

Introduction

The pandemic situation that society has faced over the past two years has been a challenge for many. For some, the imposition of restrictions on dining and entertainment was a difficult consequence. However, some people struggled with the lack of freedom to train and participate in competitions, which ultimately served as a culmination of the effort invested in physical fitness. To address these needs, the popularization of the bicycle trainer, which meets the aforementioned needs, has been addressed. Apps enabling competition and training have developed, transforming e-cycling into a sport.

Professional training often involves recording training parameters using various technologies, including heart rate monitors, pulse oximeters, real-time power measurement, and much more. The high level of accuracy of data obtained during training has created opportunities to use machine learning tools to assess the quality of preparation for specific e-cycling competitions. The study area was chosen based on the perceived opportunity to collect accurate training data from a specific group of athletes participating in identical competitions.

Cite this Article as: Bartosz OCIMEK and Joanna WIŚNIEWSKA, Vol. 2025 (34) "Machine Learning to Support the Effectiveness of Physical Training " Communications of International Proceedings, Vol. 2025 (34), Article ID 4637625, https://doi.org/10.5171/2025.4637625

This paper describes a computational experiment in which machine learning methods (see: Mahesh (2020)) were used to improve the effectiveness of sports training. As part of this project, training data from the STRAVA platform (see: Bloomberg (2020)) were collected from selected e-cycling athletes who participated in the Polish Open Championships at the PGE National Stadium in Warsaw. These data were then prepared to conduct the machine learning process using selected learning methods. Four machine learning methods were used, conducting a series of tests and collecting results, which were later used, among other things, to compare models.

The resulting summary of the above results allowed for an assessment of the data quality and the effectiveness of the created models in terms of planning and analyzing training data for the specific characteristics of e-cycling races. Conclusions were drawn regarding the research implementation in terms of training machine learning models and the results achieved.

Sport Description and Data

Virtual cycling workouts are performed using a stationary bike or a bike attached to a trainer. The trainer/stationary bike is connected to a central unit/laptop – see Fig. 1. The screen in front of the user displays a suitable application that visualizes a cycling race. One such platform is Zwift. The resistance of the bike is adjusted to the route shown on the screen, i.e., depending on the current incline, the resistance the trainer applies during the workout.

Fig 1. E-bike example (source: author's private collection)

Data Acquisition

This work utilized training data from athletes who participated in the Polish Open E-cycling Championships over a period of time. All data was collected and manually processed for the purpose of this thesis. A total of 223 athletes participated in the competition. For this purpose, 75 participants were randomly selected based on the results table from zwiftpower.com, using a pseudorandom number generation function in Excel.

The data acquisition process can be divided into several stages. The first was to correlate selected athletes (from zwiftpower.com) with their Strava accounts. This stage was not automated. The criteria adopted during this process were:

- The competitor must not have a private account, so that the data is accessible to every logged-in user.
- The competitor must have activity from February 19, 2022, in the form of the Polish Open Championship, to ensure that the account belongs to the correct competitor.

The next step was to obtain activity numbers and data (Fig. 2), which refer to individual training sessions for a specific athlete, within a given time period, i.e., 3 months preceding the Polish Open E-cycling Championships. The period is November 18, 2021 – February 18, 2022. Ultimately, the solution should be expanded to include the ability to process sensory data acquired in real time and processed on a dedicated computing server (by Chmielewski et al. (2018), (2020)) using the .NET environment and dynamic UI frameworks, as in Frąszczak (2022).

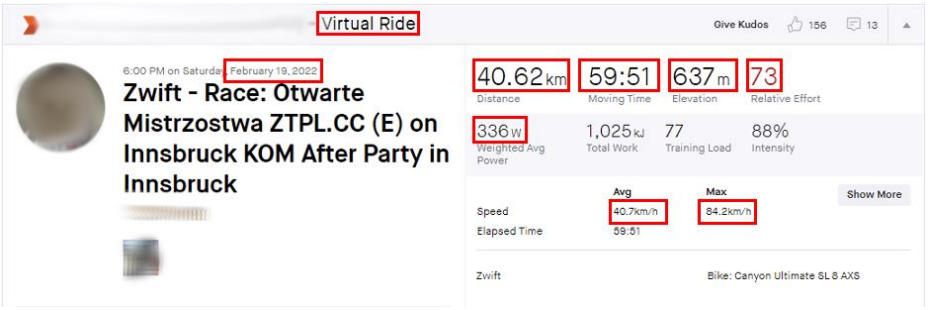

Fig 2. Data suitable for modeling

Dataset Description

The data collected was selected to provide as much information as possible about the training sessions of individual athletes. Due to the diversity of training sessions, the collected data is limited to three types of activity:

- Virtual ride,
- Standard bike ride,
- Running.

The only way to categorize activities other than those listed above would be to present them quantitatively, which could yield misleading results. If one athlete performed strength training at the gym, while another recorded a badminton workout, presenting these activities and comparing them quantitatively would be meaningless due to the different nature and purpose of the physical activity. Another reason why only three activity types were selected is that few of the selected athletes engaged in activities other than those listed above. Hence, the decision was made to limit the collected information to avoid errors/problems during the model training process.

The dataset for a single athlete contains a set of activities that were performed during a given period. Each activity consists of the following set of variables:

- id_act – activity ordinal number,
- vr – binary representation of the type of activity (0 – virtual training, 1 – standard training),
- act_type – activity type (0 – cycling, 1 – running),
- distance – the distance the athlete ran during training, regardless of the type of activity, measured in kilometers,
- time – training length shown in minutes,
- elev – total slope of the ground that the athlete covered during training, regardless of the type of activity, measured in meters,
- rel_eff – relative effort is an analysis of a user's heart rate data. By tracking their heart rate during training and its level relative to their maximum heart rate, a value is assigned to represent how strenuous the training was for a given athlete. This metric was created by Dr. Eric Bannister,
- avg_power – the power a competitor generates varies depending on terrain, slope, wind, and other factors. Average power takes all these changes into account and presents the value in watts,
- avg_speed – average speed during activity, regardless of training type, presented in kilometers per hour,
- max_speed – maximum speed achieved during activity, regardless of training type, presented in kilometers per hour.

After obtaining the above data, calculations were performed for each activity, and variables were then created that represent the training period of a single athlete. In the aggregate dataset, the athlete is described by the following variables:

- id_ath – player's ordinal number,
- avg_dist_ride – average distance covered during cycling training, regardless of whether the activity was virtual or standard, presented in kilometers,
- avg_dist_run – average distance covered during running training, presented in kilometers,
- sum_dist_ride – total distance traveled during all cycling training sessions in a given period, presented in kilometers,
- sum_dist_run – total distance traveled during all running training sessions, presented in kilometers,
- vr_val – ratio of virtual training to real-world training, range from 0 to 1,
- avg_elev_ride – average value of the incline covered during cycling training, presented in meters,
- avg_elev_run – average value of the incline covered during running training, presented in meters,
- act_count – total number of activities performed in a given period,
- act_ride – the ratio of the number of cycling training sessions to running training sessions,
- pwr_per_act – the average power generated during an activity. This is the arithmetic mean of all avg_power values for individual athletes,
- eff_per_act – the average relative effort during the activity is the arithmetic mean of all rel_eff values achieved by individual athletes,
- spd_ride – average cycling speed, expressed in kilometers per hour,
- spd_run – average running speed, expressed in kilometers per hour,
- time_ride – average time spent cycling during activity, presented in minutes,
- act_run – ratio of running training to cycling training,
- place – the competitor's place achieved during the Polish Open E-cycling Championships,
- preparation – the dependent variable, which is the classification of the 'place' variable. The places taken by the competitors were divided into 5 categories: excellent(1), very good(2), good(3), average(4), low(5). The first category includes the first 40 competitors who reached the finish line during the competition. Similarly, for every 40 places, the category decreases by one.

Variable Reduction

In every machine learning model development process, the most important step is to properly prepare the data so that the model can effectively generate predictions based on the information provided. Some of the many data preparation steps include removing duplicates (repeated records), removing empty entries (those that provide no information to the model), and removing variables that have little or no impact on the final result. The appropriate selection of explanatory variables influences the quality of the model after the training stage.

In this work, a correlation map (Fig. 3) was used to limit the number of variables, which in each cell presents the correlation coefficient ρ between two variables.

When identifying variables that are insignificant, the paper focuses on the ‘preparation’ variable and its correlations with other variables.

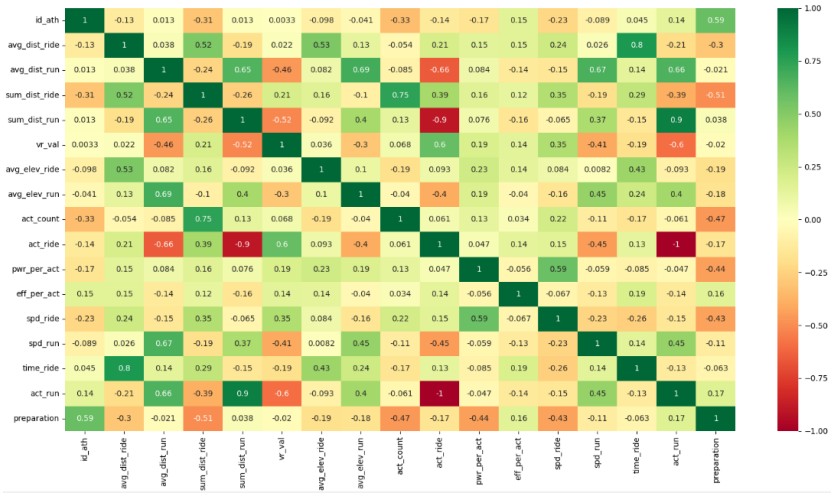

Fig 3. Variable correlation matrix

Despite the high correlation coefficient (0.59), the 'id_ath' variable was omitted due to the fact that it has no impact on the athlete's training. It is the athlete's index in the data set. To eliminate the remaining independent variables, a correlation value range (1) was created, which indicates whether the variable will be included in the model or not:

$$-0{,}1 < x < 0{,}1, \tag{1}$$

where x is the correlation coefficient of individual variables with the 'preparation' variable. After applying the above relationship, the following variables were omitted:

- avg_dist_run
- sum_dist_run
- vr_val
- time_ride

The omission of the above variables from the model indicates that, given the given set of athletes' training data, the distance covered during running training (total and average) and the ratio of home training (trainer or stationary bike) to training on a standard outdoor bike had the least impact on the level of preparation achieved for the discussed competition. Classifying the "time_ride" variable as an omitted value indicates that the time spent on a single activity has little impact on the final result, which is the level of preparation for the competition.

Overview of Results Obtained During Machine Learning Experiments

Selected Models

Selected machine learning models, both supervised and unsupervised (see: Barlow (1989)), were used to analyze the collected data. Each model used a uniform training-to-test data ratio (85% of the collected data constitutes the training set, the remaining 15% is the test set). Due to the dataset's 75-item structure, constructing validation data was not considered. The initial section of the paper describes the machine learning models, and then the results of the selected models, fed with the prepared data, are presented later in the paper.

Linear Regression

Linear regression (see: Maulud and Abdulazeez (2020)) may not be a model that should predict a competitor's place in a competition, but for comparison, the analyzed problem was also solved using this method. The dependent variable y stores information about five classes created based on the competitor's place in the competition. If it is continuous, the machine learning model, using linear regression on the dataset created for the purpose of this work, achieved an accuracy of 0.74 (using appropriate functions from the Python *sklearn* package).

The predicted value of y' was rounded to one, and in the case of 1 > y' > 0, the value of y' was assumed as y' = 1. This was done to create a confusion matrix, since linear regression applies to continuous values. The purpose of creating the matrix (Fig. 4) was to construct measures that would allow for a comparison of the quality of the machine learning models discussed in this chapter.

	1	2	3	4	5
1	4	0	0	0	0
2	0	3	0	0	0
3	0	2	1	0	0
4	0	0	1	0	0
5	0	0	0	1	0

Fig 4. Test set classification error matrix for regression

As an additional check of the quality of the prepared model, the root mean square error (RMSE) was calculated, which is represented as the square root of the MSE (mean square error). For the analyzed linear regression model, with the given data set, the RMSE was 0.63.

Classification trees

The machine learning model, using a classification tree (see: Rokach and Maimon (2009)) for the dataset created for the purpose of this work, achieved a model's accuracy of 0.5. The *sklearn* package allows visualizing the effect of the model that uses decision trees to solve a problem. Such a graphic shows the entire process, starting from the root to the leaves of the tree. Each node is detailed:

- Explanatory variables selected as part of the decision function. Denoted as X[i], where i denotes the index of the variable occurring in the dataset. A description of the assignment of individual variables is provided in Table 1,
- Gini index,
- Number of samples that were qualified for a given node (samples). The sum of all leaf samples is equal to the number of records used in the training set (63) to train the model,
- Vector indicating the class and the number of qualified samples (value).

Table 1: Error! Reference source not found.

avg_dist_ride	sum_dist_ride	avg_elev_ride	avg_elev_run	act_count	act_ride	pwr_per_act	eff_per_act	spd_ride	spd_run	time_ride	act_run
X[0]	X[1]	X[2]	X[3]	X[4]	X[5]	X[6]	X[7]	X[8]	X[9]	X[10]	X[11]

Based on the results obtained as a result of running the program, a classification error matrix was created (Fig. 5), based on which measures will be calculated for the purpose of comparing different machine learning models used in this project.

	1	2	3	4	5
1	2	1	0	0	0
2	1	3	1	0	1
3	0	2	1	0	0
4	0	0	0	0	0
5	0	0	0	0	0

Fig 5. Classification error matrix for decision trees

Artificial Neural Networks (ANNs)

In a machine learning model for ANNs (see: Jain et al. (1996), Wang (2003), Zhang (2018)), backpropagation, which operates by minimizing the sum of squared errors, was used to change the connection weights of neighboring elements in individual network layers. Since a multilayer perceptron was used to implement the model training process, the activation function was a sigmoid function. A single hidden layer containing 9 neurons was used for the machine learning stage. Using the function included in the sklearn package, which calculates the model's accuracy, the model achieved a score of 0.58. An error matrix was created based on the obtained results (Fig. 6).

	1	2	3	4	5
1	4	1	0	0	1
2	1	0	0	0	1
3	0	0	1	0	0
4	0	0	1	0	0
5	0	0	0	0	2

Fig 6. Classification error matrix for ANN

Clustering

As part of additional experiments, a clustering test was carried out on the created dataset to check whether the unsupervised machine learning model would indicate classes based on multidimensional data based on the training of individual players and whether the quantitative distribution of players in individual classes, prepared by the algorithm, would be close to the value determined manually.

For this purpose, the k-means method was used (see: Jain (2010)). The machine learning model parameters indicated the number of target clusters k=5. Fig. 7 shows two block charts representing the comparison of the number of players assigned to individual classes based on manual classification (blue) and their assignment as a result of the clustering algorithm (orange). The prepared summary shows the difference in clustering results based on "raw" and normalized data.

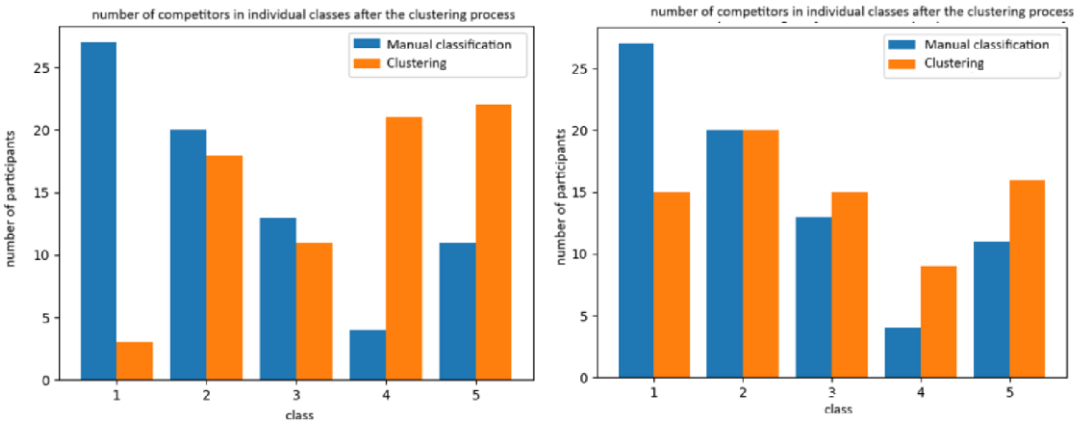

Fig 7. Summary of the cluster distribution among individual classes before (left) and after (right) the data normalization process

The discussed model results will be based on the graph on the right side of Fig. 7. The number of players assigned to the second class is the same in both cases. With the exception of the first class, the difference in the number of clusters was assigned within the acceptable limit.

As an additional test, an elbow plot was constructed to check the number of clusters indicated by the elbow method. For unnormalized data, the number of indicated clusters is $k = 3$ (Fig. 8). However, for normalized data, the number is $k = 4$.

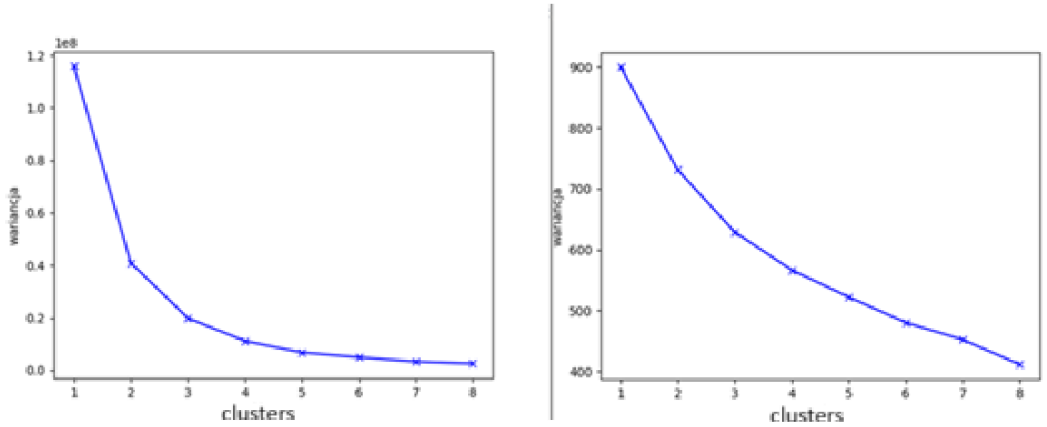

Fig 8. Comparison of elbow graphs before (left) and after (right) the normalization process

Fig. 7 and Fig. 8 demonstrate that normalizing the data before machine learning affects the effectiveness of the k-means method. This is because, after the normalization process, the model can operate on data occurring on a similar scale.

Supervised Learning Models Results Summary

While collecting the results of individual machine learning models, classification error matrices were created, which, in addition to visualizing the correctness of the operation of a given algorithm, allowed for the calculation of measures that enabled the comparison of the effectiveness of the described machine learning methods in terms of their application to the created data set.

The classification error matrix indicates the number of incorrectly and correctly classified values within the trained machine learning model. The results obtained for each model are presented in Fig. 9, Fig. 10, and Fig. 11.

	TP	TN	FP	FN	SUMA
1	4	8	0	0	12
2	3	7	0	2	12
3	1	8	2	1	12
4	0	10	1	1	12
5	0	11	1	0	12

	SENSIVITY	SPECIFITY	PRECISION
1	1	1	3
2	0.6	1	3.33
3	0.5	0.8	3
4	0	0.91	10
5	0	0.92	11

Fig 9. Calculated measures of individual classes based on the results obtained using linear regression

	TP	TN	FP	FN	SUMA
1	2	8	1	1	12
2	3	3	3	3	12
3	1	8	2	1	12
4	0	12	0	0	12
5	0	11	0	1	12

	SENSIVITY	SPECIFITY	PRECISION
1	0.67	0.89	3.33
2	0.5	0.5	1
3	0.5	0.8	3
4	0	1	0
5	0	1	0

Fig 10. Calculated measures of individual classes based on the results obtained using the classification tree

	TP	TN	FP	FN	SUMA
1	4	5	2	1	12
2	0	9	2	1	12
3	1	10	0	1	12
4	0	11	1	0	12
5	2	8	0	2	12

	SENSIVITY	SPECIFITY	PRECISION
1	0,8	0,71	1,5
2	0	0,82	4,5
3	0,5	1	11
4	0	0,92	11
5	0,5	1	5

Fig 11. Calculated measures of individual classes based on the results obtained using neural networks

Using the above table values, we calculated the sensitivity, specificity, and precision metrics for each model. A summary of the results achieved by the machine learning models is presented in Table 2.

Table 2: Error! Reference source not found.

Measure \ Method	Linear regression	Classification tree	Artificial neural networks
Sensivitiy	0.67	0.5	0.58
Specificity	0.92	0.88	0.90
Precision	4.33	4	4.17
Score (accuracy)	0.74	0.5	0.58

Conclusions

In the light of the obtained results of the discussed machine learning models, i.e. linear regression, classification tree and neural networks, it is possible to indicate the machine learning method that was most suitable for the discussed data set.

Among the presented models, the best performing model was the one using linear regression, achieving the highest values of the measures, despite the fact that linear regression is intended to predict continuous values and the output dataset was based on classified values.

The positioning of the machine learning methods in this study in terms of learning efficiency may have been influenced by the specific cutoff point, which was set as a rounding of the value to one. Had this threshold been defined more restrictively, linear regression could have achieved significantly lower results.

The lowest values, presented in Table 2, were achieved by the machine learning model using classification trees, while the model based on artificial neural networks achieved slightly better results. In both cases (neural networks, classification trees), the reasons for the low model learning results could be the fact that the models were fed with a small dataset and the number of predicted classes (five) in the output set.

Acknowledgment

The work was financed by the Military University of Technology in Warsaw, Poland as part of the project No. UGB 531-000023-W500-22.

References

- Barlow, H. B. (1989) ‘Unsupervised learning,’ *Neural Computation*, 1 (3), 295–311. doi: 10.1162/neco.1989.1.3.295.
- Bloomberg. (2020) ‘Strava said to seek new investors at 1 billion plus valuation.’ [Online]. [Accessed October 30, 2025]. Available: https://www.bloomberg.com/news/articles/2020-10-02/strava-said-to-seek-new-investors-at-1-billion-plus-valuation
- Chmielewski, M., Frąszczak, D. and Bugajewski, D. (2018) ‘Architectural concepts for managing biomedical sensor data utilised for medical diagnosis and patient remote care.’ doi: 10.13140/RG.2.2.20798.59204.
- Chmielewski, M., Frąszczak, D. and Bugajewski, D. (2022) ‘Design and development guidelines for biomedical systems capable of wearable sensor data fusion and health events reasoning: case study.’ *Figshare*. doi: 10.6084/M9.FIGSHARE.19672689.
- Frąszczak, D. (2022) ‘NEFBDAA — .NET Environment for Building Dynamic Angular Applications,’ *SoftwareX*, 19, 101163. doi: 10.1016/j.softx.2022.101163.

- Jain, A. K. (2010) 'Data clustering: 50 years beyond K-means,' *Pattern Recognition Letters*, 31 (8), 651–666. doi: 10.1016/j.patrec.2009.09.011.
- Jain, A. K., Mao, J. and Mohiuddin, K. M. (1996) 'Artificial neural networks: a tutorial,' *Computer (Long Beach Calif)*, 29 (3), 31–44. doi: 10.1109/2.485891.
- Mahesh, B. (2020) 'Machine learning algorithms - a review,' *International Journal of Science and Research (IJSR)*, 9, 381–386. [Online].
- Maulud, D. and Abdulazeez, A. M. (2020) 'A review on linear regression comprehensive in machine learning,' *Journal of Applied Science and Technology Trends*, 1 (4), 140–147.
- Rokach, L. and Maimon, O. (2009) 'Classification trees,' in *Data Mining and Knowledge Discovery Handbook*. Springer, 149–174.
- Wang, S.-C. (2003) 'Artificial neural network,' in *Interdisciplinary Computing in Java Programming*, S.-C. Wang (ed.), Springer US, Boston, MA, 81–100. doi: 10.1007/978-1-4615-0377-4_5.
- Zhang, Z. (2018) 'Artificial neural network,' in *Multivariate Time Series Analysis in Climate and Environmental Research*, Springer, 1–35.