

## RESEARCH ARTICLE

# Electric Vehicles Charging Sessions Classification Technique for Optimized Battery Charge Based on Machine Learning

SILVANA MATRONE<sup>1</sup>, (Graduate Student Member, IEEE),  
EMANUELE GIOVANNI CARLO OGLIARI<sup>1</sup>, (Member, IEEE),  
ALFREDO NESPOLI<sup>1</sup>, (Student Member, IEEE),  
GIAMBATTISTA GRUOSSO<sup>2</sup>, (Senior Member, IEEE),  
AND ALESSANDRO GANDELLI<sup>1</sup>, (Member, IEEE)

<sup>1</sup>Department of Energy, Politecnico di Milano, 20156 Milan, Italy

<sup>2</sup>Dipartimento di Elettronica, Informazione e Bioingegneria (DEIB), Politecnico di Milano, 20156 Milan, Italy

Corresponding author: Silvana Matrone (silvana.matrone@polimi.it)

**ABSTRACT** The fast increase in electric vehicle (EV) usage in the last 10 years has raised the need to properly forecast their energy consumption during charge. Lithium-ion batteries have become the major storage component for electric vehicles, avoiding their overcharge can preserve their health and prolong their lifetime. This paper proposes a Machine Learning model based on the K-Nearest Neighbors classification algorithm for EV charging session duration forecast. The model forecasts the duration of the charge by assigning the event to its correct class. Each class contains the charging events whose duration is comprised of a certain interval. The only information used by the algorithm is the one available at the beginning of the charging event (arrival time, starting SOC, calendar data). The model is validated on a real-world dataset containing records of charging sessions from more than 100 users, a sensitivity analysis is performed to assess the impact of different information given as input. The effectiveness of the model with respect to the benchmark models is demonstrated with an increase in performance.

**INDEX TERMS** Classification algorithms, electric vehicle charging, forecasting, machine learning.

## I. INTRODUCTION

When dealing with Electric Vehicle (EVs) chargers devoted to private use, the charging power profiles of a charging unit are representative of the user behavior (usually a family). Likely, the EV charger is used to charge a single vehicle, which ensures that the charging profile is similar during each charge. The main feature that differentiates charging events is their duration. Taking into account the previous considerations, it would be useful to forecast in advance the “type” (short/long, complete/incomplete) of charging event (short/long, complete/incomplete) that is going to occur at the moment the charging car is plugged in [1]. The capability of

the charging system to predict how long the EV is staying in charge mode would permit the implementation of new features in the charging control system such as the activation of an optimized battery charging [2], [3] similar to the ones that can be currently found in some smartphones. The charging system would be also capable of choosing to activate a “fast” or “slow” charge, which will have a different impact on the power load demand of the entire household. This new optimized charge would give the user a better charging experience suited to their habits, and at the same time, it would preserve the state of the battery. For instance, from IOS 13 system on, thanks to the prediction of user daily habits, the optimized charging system of an iPhone, initially charges the battery up to about 80% and then completes the charge shortly before the forecast finish time [4]. In this way, the battery is never at

The associate editor coordinating the review of this manuscript and approving it for publication was Jie Gao<sup>1</sup>.

100% charge for a long period, a condition which may cause battery degradation due to voltage exceeding the limit when the battery is fully charged [5]. The objective of the paper is to propose a supervised machine learning model capable of “predicting” the duration of the next EV charging event, happening at a certain EV charging station, by assigning the event to the correct class that is representative of its duration interval. A K-Nearest Neighbors (KNN) model is chosen to perform the classification task given only a limited amount of features as input [6]. Although the KNN has already been applied in literature [6], [7], [8], previous studies on EVs charging sessions use the KNN as a regressive prediction model; on the contrary, the present study proposes a KNN Classification [9], [10] model for classifying charging events in different duration categories based on temporal features. Moreover, the majority of the studies in the literature focus more on the energy demand prediction than on the pure charging duration prediction based on user behavior. In [11], session duration is predicted to help users in trip planning. The contributions of this paper are:

1) The proposal and the test of an application of a KNN model to EV charging session duration forecast. The KNN is used as a classification model, in contrast with the other studies in the literature that are rather based on regression. Charging events are classified into different duration categories based on temporal features.

2) The analysis of the impact of additional temporal data used as input to the model. We perform a sensitivity analysis on the employment of information such as the day of the week in which the charging is occurring and the time passed since the last charge.

The novelty of the proposed paper consists in accurately forecasting the class of EV charge power profiles in terms of a reduced number of features (duration) which is a key parameter for EV charging stations and DSOs (Distribution System Operator). This is a change of the perspective in the traditional load power curve methodology which usually relies on time-series forecasting methods. The proposed model is validated on a real-world dataset containing information about charging events that occurred in more than 100 EV charging stations around the UK. Thanks to its simplicity and speed the proposed KNN algorithm, and thanks to the possibility to generalize the model, it could be implemented and fully work locally on any type of charging towers.

The paper is organized as follows. Section II describes the methodology followed in the research, analyzing also the KNN model from a theoretical point of view, the evaluation procedure that will be used to assess the performances of the model, and, introduces the case study to which the developed model is applied. The results of the simulations are collected in Section III and then discussed in Section IV.

## II. METHODOLOGY

Figure 1 shows the workflow followed in the study. The dataset is first divided into smaller subsets, each one containing the sessions belonging to a single charging point. The

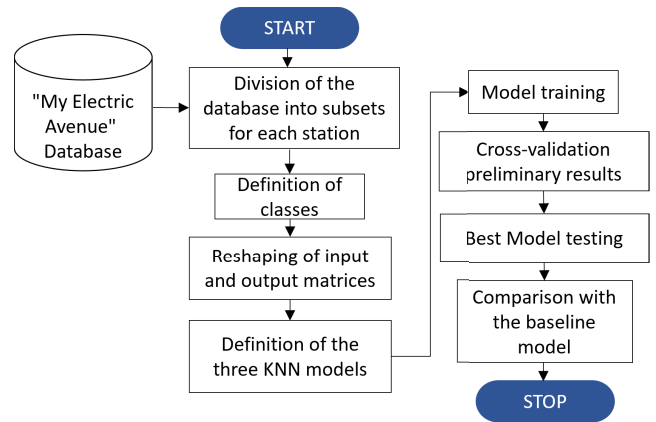


FIGURE 1. Graphical representation of the followed workflow.

definition of the classes in which charging sessions have to be categorized is done based on the duration of the charge because it is the feature that deeply influences the optimized charging. Then, the data in each subset is reshaped in order to enter the supervised machine learning KNN classification model. Three models, differing in the data given as input, are trained using the reshaped data, the best model is assessed according to cross-validation preliminary results and metrics that are described later in this paper. The model showing the best results is finally tested using the test set of data and compared to the baseline model.

### A. CASE STUDY

The data used to validate the proposed model comes from a real-world dataset containing the charging information about 200 domestic EV chargers in the UK called My Electric Avenue. “My electric avenue” is a project aimed to examine the impact of EVs on the local electricity network as sales of EVs continue to rise. Several clusters of neighbors were created in different regions in the United Kingdom. Each participant was equipped with a Nissan LEAF electric car for 18 months from 16/02/2014 to 29/11/2015, this allowed us to mimic a future scenario in which many people in an area choose to use a pure electric car or plug-in hybrid one. Over 100 families, each one assigned to one of the clusters, were successfully recruited and the usage of the respective EVs was monitored. A further 100 people were recruited for the project’s Social Trials with a total of 225. Trials participants also drove Nissan LEAFs but without the technology capable of monitoring and controlling the electricity used during the charge [12]. For the purpose of this study, the data of participants in the Social Trial are not used. Participants were divided into 10 clusters, each of them containing up to 12 based on the region they were located. The matrix regarding EV charge data schematizes 76.699 charging events, the entries of the matrix are:

- **id**: Identifier for the session record
- **ParticipantID**: Identifier of the car

- **BatteryChargeStartDate:** Date and Time of the beginning of the session
- **BatteryChargeStopDate:** Date and Time of the ending of the session
- **Starting SoC (of 12):** State of Charge of the battery at the beginning of the session
- **Ending Soc (of 12):** State of Charge of the battery at the end of the session

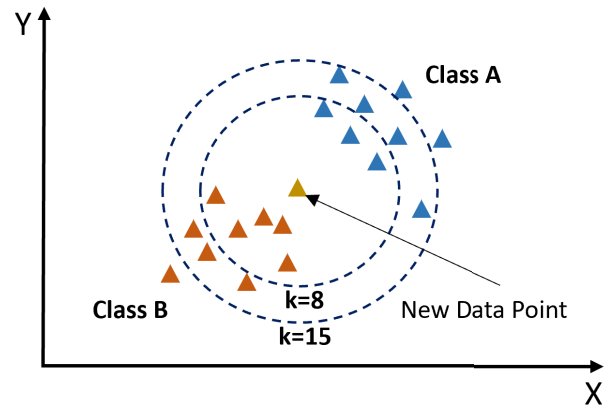
MyElectricAvenue dataset is a collection of home stations, therefore it is a good example of a situation like the one described in Section I, each charger is belonging to a family that charges always the same car, so it is a perfect way to test our model and its capability to learn the typical behavior of the users and their habits to correctly classify the type of charging each time. The prediction model can be installed directly on the charging station, it trains periodically on the data collected at each recharge to model users charging habits. When the vehicle is connected at the beginning of the charge, the information on the SoC, the date and time is available to the station and therefore to the model that can “predict” the duration of the charge based on the training performed on previous charging events. The optimized battery system is activated according to the predicted duration. At the end of the charge, the effective duration of the session is recorded to be used as new information for the next training of the model.

### B. KNN ALGORITHM

The KNN algorithm is a supervised machine learning technique used to solve both classification and regression problems. It is based on the assumption that similar things are near to each other. In the classification case, it is able to classify an unlabeled point given as input based on the class to which the  $k$  nearest labeled points belong, which are indeed its nearest neighbors. Given a training set of labeled data, a positive integer  $k$  of neighbors, and a test observation  $x_0$ , the KNN classifier first identifies the  $k$  points in the training data that are closest to  $x_0$ , forming the domain  $N_0$ . The nearest neighbors are found calculating the distance between  $x_0$  and all the points in the training set, the metric commonly used is Minkowski with  $p = 2$ , equivalent to the standard Euclidean metric. It then estimates the conditional probability for  $x_0$  being part of class  $j$  as the share of points in  $N_0$  belonging to class  $j$ . As described by Equation (1), the probability of  $x_0$  being assigned to class  $y = j$  is equal to the ratio of the number of points  $I$  in  $N_0$  that belong to  $j$  and the total number of neighbors  $k$ . Finally, it proceeds to classify the test observation  $x_0$  to the class with the largest probability. Figure 2 shows the functioning of KNN algorithm [13]:

$$Pr(X = x_0|y = j) = \frac{1}{k} \sum_{i \in N_0} I(y_i = j) \quad (1)$$

The number of  $k$  has a drastic effect on the KNN classifier obtained. As  $k$  grows, the method becomes less flexible and the decision boundaries are closer to a linear function. A decision boundary is a curve, characteristic to the KNN



**FIGURE 2.** Graphical representation of KNN classification of a point for two different  $k$  values. When  $k$  is set equal to 8 the new point is assigned to class B, when  $k$  is set to 15, the point is classified as A.

model and  $k$  value, that can be ideally drawn in the plan in which the data points are represented, all the points on one side of the curve will be assigned to a certain class, the ones on the other side to a different label. However, with increasing  $k$ , predictions become more stable until eventually reaching a maximum and starting to decrease again.

### C. DATA PROCESSING

Analyzing the data, it was noticed that the majority of the sample charging events have a duration lower than or equal to 1 hour, few of them last more than 8 hours, and a large group has a correspondent time interval between 2 and 6, with a mean value of 3 hours circa. Based on the last considerations and the results of a previously performed sensitivity analysis, the best results were shown when dividing the dataset into 3 different classes. The presence of more than 3 classes caused a decrease in accuracy due to the lower population of each class in the training set. This decision makes the problem a multi-class classification since there are more than 2 classes. The first class contains the sessions with a duration lower than 2 hours, the second includes the session between 2 and 6 hours, and the last has all the rest of the events that last more than 6 hours. The classification described is repeated for each of the 110 participants.

Before using the KNN algorithm introduced in Section I, data has to be properly reshaped. The input matrix  $X$  has shape  $(n_{samples}, n_{features})$ , where  $n_{samples}$  is the number of sessions belonging to a unit and the features that are the information that the charging system sees the moment the EV plugs in, as per Section I they will be different in the three cases that will be analyzed and reported in Table 1.

The first model is trained using only the Starting SoC and the time of the day in which the charge is starting. The starting SOC, if properly evaluated by the Battery Management System (BMS) [14], [15] is a crucial feature in the prediction of the duration of the charge assuming that it is brought to completion. The time of the day gives the model information about the habits of the user. In the second model

TABLE 1. Study cases description.

Case	Input data
Case 1	Starting SoC, hour of the day, minute
Case 2	Starting SoC, hour of the day, minute, day of the week
Case 3	Starting SoC, hour of the day, minute, day of the week, time passed since the last charge

also information about the day of the week is added, which could be useful for the model to learn what type of charge is more common during the weekend rather than on working days. The last model is trained also using the time passed since the last charging. These three cases are summarized in Table 1.

The output matrix  $y$  has a similar shape with  $(n_{samples}, label)$ , where the label is the number of the class in which the session in input has been classified, the classifier tries to predict the duration of recharge by understanding in which class it falls. Based on the predicted class, the charging system will start a charge that is suited for the available time interval. The input and output data are split into a training set and a testing set. Considering the low number of samples available for each charging unit, 90% of the samples are used for the training while the remaining 10% is used to test the model.

The same KNN architecture is used for all of the 3 cases presented in Table 1. The KNN algorithm is really simple and needs a few parameters to perform a classification task. In our case, a KNN classifier with uniform weights was implemented. The uniformity of weights makes it so that, when assessing the most probable class for an input data point, the model considers all the neighbor points to have the same weight, which means their influence on the final classification is not depending on their distance from the input point but is constant for all of them. The distance metric used by the model to find the neighbors of a point is the Euclidean one (Minkowski with  $p = 2$ ). The last parameter needed by the model is the number of neighbors to find and take into account for the choice of a class.

The  $k$  number of neighbors has a strong influence on the results of the classification problem, for this reason, the KNN model has been trained and tested on each subset of data containing the sessions of a charging station for each number of neighbors ( $k$ ) from 1 up to 25. The optimal value of  $k$  that maximizes the accuracy for each charging station has been individuated. All following calculations are done using the said optimal value. In general, was not possible to identify a rule in the value of  $k$  that maximizes accuracy in the proposed problem optimal  $k$  assumes every value between 1 and 22 with no apparent principle. The model, trained with the optimal value of  $k$  for each station, is then used to make predictions on the test set.

D. MODEL EVALUATION

The evaluation of the quality of any classifier usually passes through the most immediate and intuitive metric that can be used: accuracy. Unfortunately, accuracy is not capable of tak-

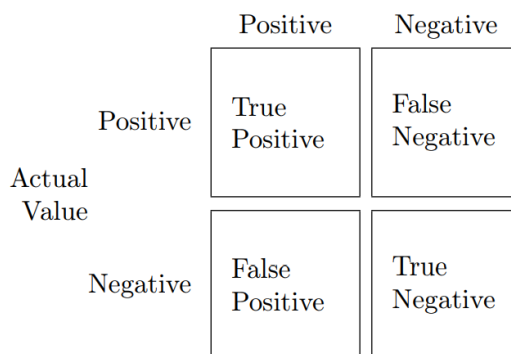


FIGURE 3. Confusion Matrix.

ing into account certain factors, for example, the possibility of having a ‘skewed’ dataset, in which the majority of the data belongs to the same class with few exceptions, it means that classifying all the samples in the same class will result in high accuracy, even if the model is not capable of detecting the other two classes. It is more representative to build and observe the confusion matrix of the classifier. Each row of the confusion matrix represents the instances in an actual class while each column represents the instances in a predicted class, in this way, it is possible to identify, the number of True Positives (TP) (the labels that were assigned correctly), True Negatives (TN) (the cases in which the classifier recognizes that a sample *is not* in a certain class), False Positives (FP) (the classifier assigns the wrong class) and False Negatives (FN) (the classifier assesses incorrectly that an item *is not* in a certain class). In a perfect classification the only non-null values are on the diagonals of the matrix representing True Positives and True Negatives [16]. The layout of a confusion matrix is shown in Figure 3 In multi-class classification, big matrices can be difficult to analyze, it is more convenient to extract concise metrics such as precision, recall, and F1. Precision is the accuracy of the positive predictions, its formula is shown in Equation 2, where TP and FP are respectively, the number of true positives and false positives. Recall (also called sensitivity) is the ratio of positive instances that are correctly detected by the classifier, Equation (3) shows how it is calculated. A combination of precision and recall called F1 is a useful metric employed in the comparison of classifiers, it is calculated as the harmonic mean of precision and recall (Equation (4)). The harmonic mean gives more weight to small values so the F1 will be high only if both precision and recall are high and similar.

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

$$F1 = \frac{TP}{TP + \frac{FN+FP}{2}} \tag{4}$$

The Receiver Operating Characteristic (ROC) curve is a graphical representation of the relationship between sensitivity and specificity. ROC curve plots the True Positive Rate (also Recall or sensitivity) against the False Positive Rate (FPR), which is the ratio of negative instances that are incorrectly classified as positive. Each point of the curve represents a couple of TPR/FPR values computed for a certain threshold between 0 and 1. Decreasing the threshold will increase the number of false positives, and therefore the degree of sensitivity. A pure random classifier just produces a straight ROC line that bisects the plan, while for a good classifier, the curve is moving toward the top left corner of the plan. The optimal threshold is the one that maximizes the TPR. The Area Under the ROC Curve (AUC), is a performance indicator for classifiers, it measures the entire two-dimensional area under the entire ROC curve from 0.0 to 1.0, indeed it is equal to 0.5 for a random classifier and equal to 1 for a perfect one. The higher the AUC, the more a model is able to distinguish between classes. Thanks to its conciseness, the AUC is effectively used to compare the performances of different classifiers. The confusion matrix, precision, recall, and the F1 score are calculated using Cross Validation, it is a K-fold validation, meaning the training set of data is split into K folds (k subsets) of equal size, then predictions are made and evaluated them on each fold using a model trained on the remaining folds. In this way, we can have a preliminary evaluation of the classifier without touching the test set. This will be used to compare three different models, they only differ in the number of information given as input to the model to train. After assessing which is the best classifier among the analyzed ones, we will finally test its performance on the test set of the original data to calculate its accuracy. To have an objective point of view on the results coming from the KNN algorithm, they are compared to the results of a baseline model such as the Zero Rule model. It assigns to each sample in the test set the label that is more frequent in the training set, for instance, if a training has 80% of samples belonging to Class 1, the zero rule algorithm will simply assign the label 1 to each session sample given in the test. This mimics perfectly a mistake that is likely to happen when making predictions with a classifier that has been trained on a set of data in which one class is much more frequent than the other (skewed).

### III. RESULTS

#### A. COMPARISON OF THE 3 CASES

The preliminary results from the cross-validation analysis allow us to evaluate the performances of each one of the models trained based on the 3 cases presented in II-A before performing the final assessment on the test dataset. Table 2, 3, 4, summarize these results showing the evaluation metrics calculated for a subset of charging stations taken as examples and their average values on all of the stations.

Looking at the tables, it is simple to notice how the average values are maximum for Case 1 and decrease over Case 2 and

**TABLE 2. Cross Validation preliminary results on Case 1.**

Station	Accuracy	Precision	Recall	F1	Number of Sessions
Average	0.69	0.56	0.62	0.58	-
CC01	0.81	0.59	0.70	0.61	84
CC02	0.89	0.49	0.53	0.51	45
CRG02	0.80	0.73	0.76	0.73	447
CRG08	0.70	0.59	0.64	0.61	387
GC03	0.74	0.61	0.69	0.62	153
GC09	0.76	0.51	0.70	0.59	85
JD01	0.61	0.52	0.57	0.54	580
JD05	0.73	0.50	0.67	0.57	113
JD07	0.97	0.90	0.91	0.91	186
MC09	0.68	0.51	0.53	0.52	259

**TABLE 3. Cross Validation preliminary results on Case 2.**

Station	Accuracy	Precision	Recall	F1	Number of Sessions
Average	0.64	0.56	0.57	0.56	-
CC01	0.75	0.54	0.55	0.54	84
CC02	0.67	0.47	0.47	0.47	45
CRG02	0.82	0.72	0.72	0.72	447
CRG08	0.72	0.63	0.64	0.63	387
GC03	0.77	0.69	0.68	0.69	153
GC09	0.71	0.70	0.69	0.70	85
JD01	0.57	0.55	0.53	0.54	580
JD05	0.69	0.56	0.59	0.57	113
JD07	1.00	0.94	0.95	0.95	186
MC09	0.75	0.66	0.65	0.59	259

**TABLE 4. Cross validation preliminary results on Case 3.**

Station	Accuracy	Precision	Recall	F1	Number of Sessions
Average	0.55	0.53	0.53	0.53	-
CC01	0.82	0.65	0.68	0.66	84
CC02	0.67	0.62	0.62	0.62	45
CRG02	0.77	0.75	0.75	0.75	447
CRG08	0.74	0.62	0.60	0.61	387
GC03	0.70	0.68	0.68	0.68	153
GC09	0.82	0.62	0.61	0.62	85
JD01	0.72	0.66	0.65	0.65	580
JD05	0.74	0.62	0.62	0.62	113
JD07	0.86	0.83	0.80	0.81	186
MC09	0.74	0.53	0.54	0.53	259

Case 3, however, to compare the three of them and assess the best one, we also considered the AUC. Before showing the results, it is important to stress that, since we are treating a multi-class classification and since the AUC evaluation is only defined for binary classification, we followed a “One vs the Rest” (OvR) approach to extend the calculations to our case, in which three different classes are present. In turn, we consider each class as positive and the others as negative, transforming the problem into a binary classification. We calculate in this way three values of AUC, one for each class when considered positive. Table 5 reports the mean values over the entire dataset of the AUC in the three study cases and the three classes. Figure 5 shows an example of the ROC curves that result from the OvR evaluation on participant “JD04” in Case 1. Figure 4 shows how the accuracy is changing along the three examined cases for the charging stations considered as examples.

Similarly to the observations made in the previous tables, also according to the AUC value, Case 1 appears to be the

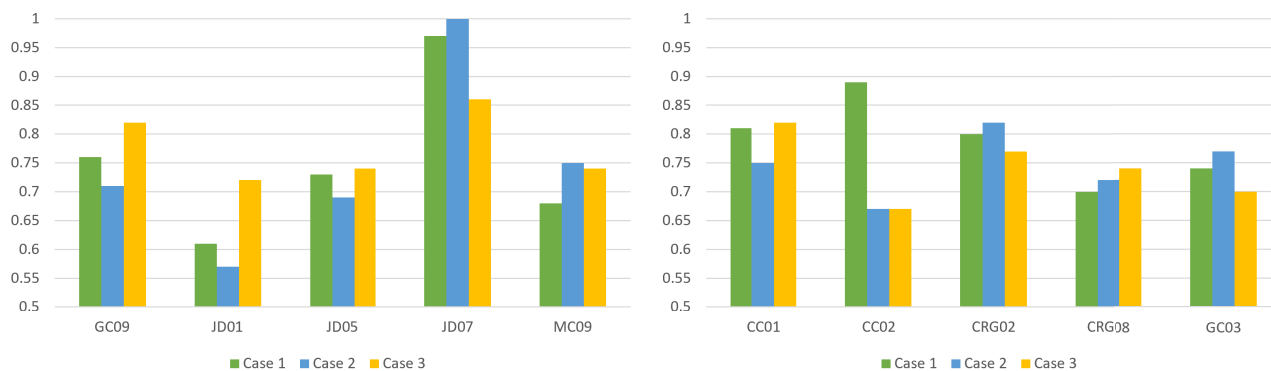


FIGURE 4. Trends of the KNN model accuracy as a function of the 3 examined cases.

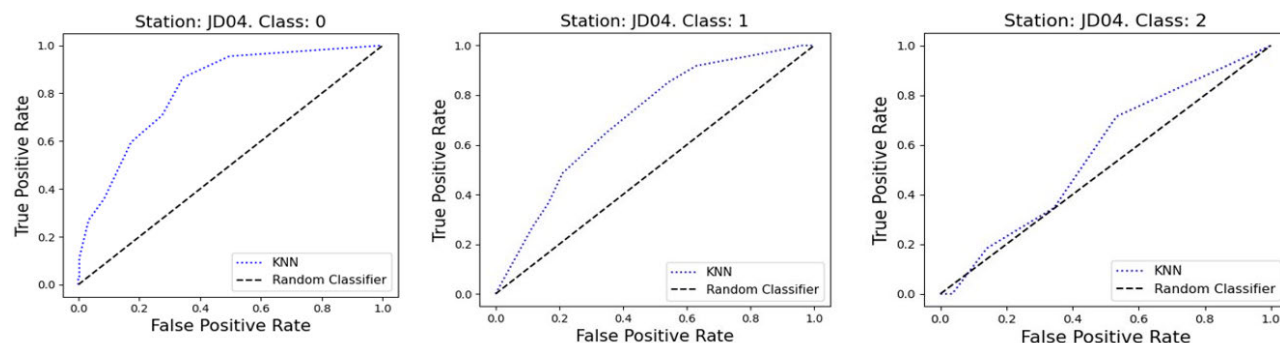


FIGURE 5. ROC curves resulting from OvR validation procedure for participant JD04 (taken as example) in Case 1.

TABLE 5. AUC average values on the entire dataset for each of the study cases and classes.

	Case 1	Case 2	Case 3
Class 0	0.61	0.57	0.54
Class 1	0.58	0.55	0.52
Class 2	0.56	0.54	0.56

best-performing one. This result is particularly interesting. Despite the fact of having more features in input, neither Case 1 nor Case 2 manage to overcome Case 1, on the contrary, they perform less. It means the information about the day of the week in which the charging is occurring and the time passed since the last charge is not capable of improving the classification accuracy, it is probable that the only effect they achieve is to complicate the input excessively, making it more difficult to learn for the model, moreover, the population of each subset of data is limited, making it difficult for the model to learn more features at once. However this is not always the case, there are particular charging stations in which the introduction of new features in the input is increasing the performances both in terms of accuracy and in terms of AUC, for instance in ‘JD07’ where accuracy is even reaching a value of 1 in Case 2. It is likely that the owners of these stations are the ones that have more repetitive behavior and habits, making this type of model more effective.

**B. TESTING OF THE BEST CASE**

Finally, we can test the best classifier on the test set, calculating its accuracy and comparing it with the accuracy of the Zero Rule benchmark model to check its improvement. Table 6 shows the results for a subset of charging stations taken as examples, in the first row are reported the average values on the entire dataset. A graphical comparison between the accuracy of the KNN model and the Zero Rule algorithm is shown in Figure 6. The model that we have built reaches an accuracy of 61% on the completely new testing data with an average improvement with respect to the Zero Rule of 69%. Looking at the single stations, in some cases, the improvement is really high (e.g. CC02), while in other cases it is a lot lower (e.g.CRG02). This behavior can be easily explained by noticing how greater improvements are found for the charging station in which the presence of the three classes is well balanced, in these cases the Zero Rule has really low performances causing the number of labels ‘0’, ‘1’ and ‘2’ is similar, hence the KNN classifier can be exploited at its full capability of predicting user behavior. On the contrary, the stations with a strong prevalence of one of the classes, the skewed ones, have smaller improvements because the Zero Rule was already predicting only the most common class as in the case of CRG02 in which the majority of data is belonging to Class 2.

TABLE 6. Results on the test set for Case 1.

Station	Accuracy	Zero Rule accuracy	Improvement
Average	0.61	0.36	69%
BL01	0.65	0.5	23%
CC02	0.6	0.067	88%
CRG01	0.57	0.19	67%
CRG02	0.94	0.88	6%
CRG08	0.98	0.86	18%
GC01	0.46	0.06	87%
MC05	0.65	0.45	30%

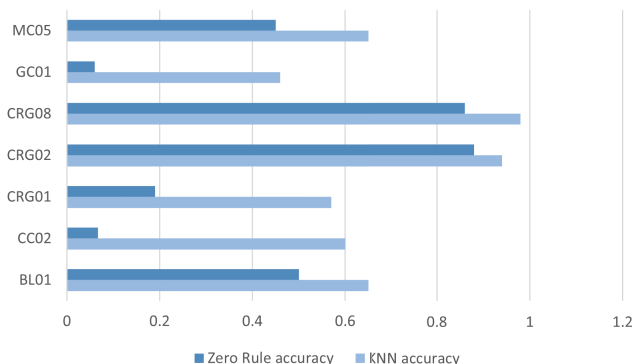


FIGURE 6. Comparison between the accuracy of the KNN model and the Zero Rule algorithm.

#### IV. CONCLUSION

In this paper, a KNN classification model has been proposed and applied to the real-world MyElectricAvenue EV charging sessions dataset. The model has proven its capability to learn the user charging habits and behavior, and then classify the charging events based on their duration. The information on session duration could play a relevant role in an optimized battery charging strategy devoted to the planning of EV charges and limiting battery damage. Three different configurations of the model have been proposed to assess the impact that different features given in input could have on the results of the classification. The two additional features (day of the week and time passed since last charging) were not capable to improve the performances of the classification model during the cross-validation, the best results came from the simple input containing just the time and starting SoC at the beginning of the charge. During the final test, the model demonstrated being capable of correctly classifying the charging event on limited input data and therefore forecasting the duration of the charging event with high accuracy. The improvement with respect to the benchmark model is even higher when the user behavior is variable in terms of the length of the charging events. The comparison of the proposed model with others present in the literature would not be significant since the latter are of the regressive type as in [11]. The error metrics used to evaluate a regression model (e.g. MAPE) are not applicable to a classification model such as the one described in our paper. For future works, we could implement a new multi-output classification

KNN model capable of classifying both cases based on the estimated duration and on the possibility that the user would stop the charging before its completion.

#### REFERENCES

- [1] S. Shahriar, A. R. Al-Ali, A. H. Osman, S. Dhoun, and M. Nijim, "Prediction of EV charging behavior using machine learning," *IEEE Access*, vol. 9, pp. 111576–111586, 2021.
- [2] C. Chen, Z. Wei, and A. C. Knoll, "Charging optimization for Li-ion battery in electric vehicles: A review," *IEEE Trans. Transport. Electrific.*, vol. 8, no. 3, pp. 3068–3089, Sep. 2022.
- [3] Q. Lin, J. Wang, R. Xiong, W. Shen, and H. He, "Towards a smarter battery management system: A critical review on optimal charging methods of lithium ion batteries," *Energy*, vol. 183, pp. 220–234, Sep. 2019.
- [4] (Oct. 24, 2022). *About Optimized Battery Charging on your iPhone*. [Online]. Available: <https://support.apple.com/en-us/HT210512>
- [5] F. Ghassani, M. Abdurrohman, and A. G. Putrada, "Prediction of smartphone charging using K-nearest neighbor machine learning," in *Proc. 3rd Int. Conf. Informat. Comput. (ICIC)*, Oct. 2018, pp. 1–4.
- [6] M. Majidpour, C. Qiu, P. Chu, R. Gadh, and H. R. Pota, "Fast prediction for sparse time series: Demand forecast of EV charging stations for cell phone applications," *IEEE Trans. Ind. Informat.*, vol. 11, no. 1, pp. 242–250, Feb. 2015.
- [7] S. Boddia and P. Agnihotri, "Comparative assessment of machine learning regression methods for power system state estimation," in *Proc. Symp. Power Electron. Renew. Energy Syst. Control*, in Lecture Notes in Electrical Engineering, S. Mohapatro and J. Kimball Eds. Singapore: Springer, 2021, pp. 391–402.
- [8] I. Ullah, K. Liu, T. Yamamoto, M. Zahid, and A. Jamal, "Electric vehicle energy consumption prediction using stacked generalization: An ensemble learning approach," *Int. J. Green Energy*, vol. 18, no. 9, pp. 896–909, Jul. 2021.
- [9] C. Fan, F. Xiao, and S. Wang, "Development of prediction models for next-day building energy consumption and peak power demand using data mining techniques," *Appl. Energy*, vol. 127, pp. 1–10, Aug. 2014.
- [10] Y. E. Yousif, "Weather prediction system using KNN classification algorithm," *Eur. J. Inf. Technol. Comput. Sci.*, vol. 2, no. 1, pp. 10–13, Feb. 2022.
- [11] I. Ullah, K. Liu, T. Yamamoto, M. Zahid, and A. Jamal, "Prediction of electric vehicle charging duration time using ensemble machine learning algorithm and Shapley additive explanations," *Int. J. Energy Res.*, vol. 46, no. 11, pp. 15211–15230, Sep. 2022. [Online]. Available: <https://onlinelibrary.wiley.com/doi/pdf/10.1002/er.8219>
- [12] *My Electric Avenue Project Summary*, EA Technol., Capenhurst, U.K., 2017.
- [13] A. Zhang, C. Z. Lipton, M. Li, and J. A. Smola. (Jul. 2021). *Dive Into Deep Learning*. [Online]. Available: <https://D2L.ai> and <https://github.com/d2l-ai/d2l-en/>
- [14] G. Gruosso, G. Storti Gajani, F. Ruiz, J. D. Valladolid, and D. Patino, "A virtual sensor for electric vehicles' state of charge estimation," *Electronics*, vol. 9, no. 2, p. 278, Feb. 2020.
- [15] D. Pau, D. Denaro, G. Gruosso, and A. Sahnoun, "Microcontroller architectures for battery state of charge prediction with tiny neural networks," in *Proc. IEEE 11th Int. Conf. Consum. Electron. (ICCE-Berlin)*, Nov. 2021, pp. 1–6.
- [16] A. Géron, *Hands-On Machine Learning With Scikit-Learn, Keras, and TensorFlow*, 2nd ed. Sebastopol, CA, USA: O'Reilly Media, Sep. 2019.



**SILVANA MATRONE** (Graduate Student Member, IEEE) received the B.Sc. degree in mechanical engineering from the University of Naples Federico II, in 2019, and the M.Sc. degree in energy engineering from Politecnico di Milano, in 2022, where she is currently pursuing the Ph.D. degree with the Electrical Division, Energy Department. Her research interest includes machine learning techniques, with applications in energy dispatching systems.



**EMANUELE GIOVANNI CARLO OGLIARI** (Member, IEEE) received the M.Sc. degree in electrical engineering, and the Ph.D. degree in electrical engineering from Politecnico di Milano, Italy, in 2016. He has been working on photovoltaic power plant design and their optimization, since 2010, and RES expected power by means of computational intelligence techniques, since 2012. He is currently a Senior Assistant Professor with the Department of Energy, Politecnico di Milano,

where he teaches with the Electrical Engineering and Photovoltaic-Based Systems Laboratory. He has collaborated with international universities and research centers (The University of Danang—University of Science and Technology, Ben Gurion University of the Negev, and Global Energy Interconnection Research Institute Europe). In 2013, he won the joint laboratories between Italy and Israel on Solar Energy ILSE.



**ALFREDO NESPOLI** (Student Member, IEEE) received the B.Sc. and M.Sc. degrees in energy engineering from Politecnico di Milano, in 2016 and 2018, respectively, where he is currently pursuing the Ph.D. degree. Since then, he was a Temporary Researcher with the Electrical Division, Energy Department, Politecnico di Milano. His research interests include AI and renewable energy sources integration in the energy systems.



**GIAMBATTISTA GRUOSSO** (Senior Member, IEEE) was born in 1973. He received the B.S. and M.S. degrees in electrical engineering and the Ph.D. degree in electrical engineering from Politecnico di Torino, Italy, in 1999 and 2003, respectively. From 2002 to 2011, he was an Assistant Professor with the Department of Electronics and Informatics, Politecnico di Milano, where he has been an Associate Professor, since 2011. He is the author of more than 80 papers in journals and conferences on the topics. He does research in electrical engineering, electronic engineering, and industrial engineering. His research interests include electrical vehicles transportation electrification, electrical power systems optimization, the simulation of electrical systems, digital twins for smart mobility, factory and city, and how they can be obtained from data.



**ALESSANDRO GANDELLI** (Member, IEEE) received the M.Sc. and Ph.D. degrees in electrical engineering from Politecnico di Milano. In 1990, he joined as the Faculty Member of Engineering with the Politecnico di Milano, where he is currently an Associate Professor in electrical engineering. From 1998 to 2001, he was a member of the Academic Senate. His research interests include the basic theory of electrical systems and electricity market. He has published more than 100 papers and three textbooks in the field of electrical and electronics engineering. He is a member of ACM, SPIE, IMAPS, and the New York Academy of Science.

• • •