energies

MDPI

Article

Synthetic Data Generator for Electric Vehicle Charging Sessions: Modeling and Evaluation Using Real-World Data

Manu Lahariya [1,*], Dries F. Benoit [2] and Chris Develder [1]

[1] IDLab, Ghent University – Imec, Technologiepark Zwijnaarde 126, 9052 Ghent, Belgium; chris.develder@ugent.be
[2] Center for Statistics, Ghent University, Tweekerkenstraat 2, 9000 Ghent, Belgium; dries.benoit@ugent.be
* Correspondence: manu.lahariya@ugent.be

Received: 15 July 2020; Accepted: 12 August 2020; Published: 14 August 2020

Abstract: Electric vehicle (EV) charging stations have become prominent in electricity grids in the past few years. Their increased penetration introduces both challenges and opportunities; they contribute to increased load, but also offer flexibility potential, e.g., in deferring the load in time. To analyze such scenarios, realistic EV data are required, which are hard to come by. Therefore, in this article we define a synthetic data generator (SDG) for EV charging sessions based on a large real-world dataset. Arrival times of EVs are modeled assuming that the inter-arrival times of EVs follow an exponential distribution. Connection time for EVs is dependent on the arrival time of EV, and can be described using a conditional probability distribution. This distribution is estimated using Gaussian mixture models, and departure times can calculated by sampling connection times for EV arrivals from this distribution. Our SDG is based on a novel method for the temporal modeling of EV sessions, and jointly models the arrival and departure times of EVs for a large number of charging stations. Our SDG was trained using real-world EV sessions, and used to generate synthetic samples of session data, which were statistically indistinguishable from the real-world data. We provide both (i) source code to train SDG models from new data, and (ii) trained models that reflect real-world datasets.

Keywords: smart grid; electric vehicle; synthetic data; exponential distribution; Poisson distribution; Gaussian mixture models; mathematical modeling; machine learning; simulation

1. Introduction

The growth of electric vehicles (EVs) in the past decade has induced significant modifications in city-wide electric grids. More than one million plug-in EVs were registered in Europe in 2018, and multiple charging stations have been installed to facilitate this growth. This rise provides opportunities to collect EV session data and use it to exploit flexibility, balance load and create responsive grids. Companies can use the data generated from charging stations to understand consumer behavior, provide incentives and make pricing decisions.

Session data collected from city-wide EV charging stations can be used for both academic and industrial purposes: the increased inflow of data has huge impacts on the energy informatics field [1]. Previous studies of different EV datasets include (i) statistical analyses of data collected in the Netherlands by ElaadNL [2,3], (ii) analysis of energy consumption of EVs on data collected by the US department of energy [4] and (iii) multiple studies on the socioeconomic effects of switching to EVs in day to day use [5,6]. However, studies require reliable session data for understanding behaviors and exploring flexibility. The scarcity of reliable data has been discussed previously [7], and its necessity has been pointed out for further research purposes. Where data are available, they

may still be protected under confidentiality by private data collectors, and not freely available for academic or public use. The lack of availability and difficulty in accessibility of EV charging session data poses a significant hurdle to further research in the field.

1.1. Related Work

EV session data contains the session duration and charging requirements of each EV. Previous studies studying the flexibility provided in the power grid [8], and in individual sessions [9], offer a statistical modeling methodology with which to understand EV sessions. Arrivals of EVs can be considered as events on a time scale, where session duration and charging load are dependent on each EV arrival event.

A probabilistic time series model using a generative adversarial network (GAN) has been used previously to generate synthetic samples in [10]; they modeled energy consumption for users. However, consumption can be represented as a continuous time series, which is not the case when we consider *EV arrivals* as discrete events in time. Another method used to model data was implemented and validated in [11]; they used a Markov chain model to generate load profiles only in individual charging stations, based on a Swedish dataset. This does not satisfy the need to model EV arrivals jointly for a set of charging stations. Statistical characterization of the session plug in times was also explored: Flammini et al. [12] used beta mixture models to represent the multi-modal distributions. They analyzed the distribution of arrival times during the day, but did not provide a synthetic sample generation process that includes a temporal component. Statistical representation of EV arrivals throughout the day using GMMs can also be used to randomly sample arrivals, e.g., in [3], for which they took data for 221 EVs to create day long profiles. Other methods include using a stochastic simulation methodology to generate a schedule of EVs for a population [13]. Aforementioned works only implemented temporal modeling on continuous time series collected from smart grids, which is not the case with arrival times of EVs. Arrival times of EVs are discrete events in time, and hence difficult to model.

The *departure time* of EV is dependent on the arrival time, so the connection times become conditional on arrivals. Departure time modeling has been explored exhaustively in [14], for both uni-modal and multi-modal data distributions. The underlying assumption is that in the 24 h duration, the probability of the event occurring is a time-varying function. A mixture of multiple distributions can be used to estimate this function. For EV connection times, these conditional probability distributions have been modeled using Abe–Lay mixtures [15], and a cylindrical WeiSSVM distribution [16]. Both Abe–Ley mixtures and the WeiSSVM distribution offer good alternatives for initializing the number of mixtures and their properties. Beta mixture models have also been used; an estimation method was suggested in [12] to estimate the departure profiles. However, generation and evaluation of samples from these mixtures were not included. The dependency of connection times on arrival times introduces a complexity that has not been addressed so far.

For predicting *charging demand*, a k-nearest neighbors algorithm was evaluated in [17], to predict the charging requirements of EVs at individual charging stations. However, it did not include the effect of EV session durations. Other methods including auto-regressive models [18] have also been explored for smart grids datasets, which can be used to synthetically generate smart meter data. A combination of arrival times, departure times and charging requirements of EVs have not been studied, and modeling them together provides an opportunity to generate synthetic samples of EV session data.

1.2. Contribution

In this paper, we present a state of the art model for generating samples of EV session data that will generate synthetic samples of (i) arrival times, (ii) connection times and (iii) charging load, for each EV. We describe this model as synthetic data generator (SDG), as defined in our previous work [19]. This includes temporal statistical modeling of arrivals and modeling of conditional distributions

for departures and the energy required for charging the EV. This differs from [3], in the sense that we generate data on each session level, whereas they have only studied charging matrices. Herein, we also define and release trained parametric SDG models that can be used to generate session data, which were not provided in [3]. In comparison to [11], wherein load profiles were modeled using a spatial Markov chain model for five charging stations, our study includes temporal modeling of EV sessions arrivals for the joint set of multiple charging stations, derived from a large-scale real-world dataset comprising about 2000 charging stations. Along with this, we also include methods to jointly model the arrival and departure times of EVs for a large number of charging stations. Compared to [12], where the arrivals of EVs were characterized for weekends and weekdays, we propose a modeling method that can be used for any set of days that have similar properties, and adopt different statistical models. Our approach also gives us further insights into consumer behavior, by providing us the rates of EV arrivals for different hours, days and months. These generated arrivals will be used to generate the departures and required energy for each session. Our main contributions from this paper include:

- A novel approach to generating synthetic data for EV sessions over a group of charging stations defined as the SDG (Section 2).
- Training of the SDG using a real-world dataset. An analysis of statistical properties of real-world data is also included (Sections 3 and 4).
- Generation of synthetic samples, and evaluation of similarity with the real-world data. We compare results from different models that can be used in SDG (Sections 5–7).
- Trained models and code are provided in GitHub (https://github.com/mlahariya/EV-SDG). Python was used for the models developed in this article (see Appendix A);

2. Modeling Methodology

We define the synthetic data generator (SDG) in this section. We define a parametric model (SDG) that can be used to generate synthetic samples of EV session data, and its inputs. We assume that each session can be described using three parameters: (i) arrival time (t_a), (ii) connection time (t_c) and (iii) required energy (E). The departure time can be calculated using $t_d = t_a + t_c$. E represents the charging load that an EV has requested (based on measured charging power throughout the full session). Session parameters for date d can be generated using Equations (1)–(3).

$$t_a = AM(d) \tag{1}$$

$$t_c = MM_c(t_a, d) \tag{2}$$

$$E = MM_e(t_a, d) \tag{3}$$

In what follows, we define (i) the arrival model (AM), (ii) the mixture model for connection times (MM_c) and (iii) the mixture model for required energy (MM_e). Trained SDG models can be used to generate a sample of data. Data generation is a two step process.

Step 1. Arrivals: We generate the arrival of EVs (t_a) for all dates in the input horizon. This horizon is the period of time for which the data needs to be generated, and can be defined using the first date (starting date) and the last date (ending date) of this period.

Step 2. Connected time and energy required: Once we have the arrivals of EVs, we generate the connected time (t_c) and energy required (E) for that particular EV arrival.

AM, MM_c, MM_e is trained for a set of dates (**S**). Dates present in **S** will have similar daily properties (e.g., arrival profiles), and we can define **S** by assuming a grouping criteria for days, e.g., we can assume that each month will have similar arrival profiles, i.e., the grouping criteria for dates is months m. For each month m, all dates of that month will be the elements of set **S**. Details about defining **S** in practice, in particular for a real-world dataset, are included in Section 4.

2.1. Arrival Models

Arrivals of EVs in a group of charging stations (poles) can be considered as events over time. For a large number of poles, we can assume that the inter-arrival times (IATs,Δt) of EVs follow an exponential distribution (which we validate in Section 4.2). Based on this assumption, one method to model arrival times of EVs is to model the time in between arrivals (Δt). A second method is to model the total number of EV arrivals in a time interval. Both these methods are defined below.

2.1.1. Inter-Arrival Time Models

To model inter-arrival times (Δt) we use the exponential distribution, which is characterized by a rate parameter λ (rate of EV arrivals). Inter-arrival time (IAT) models are defined as follows:

$$t_i = t_{i-1} + \Delta t \tag{4}$$

$$PDF(\Delta t) = \lambda_{i-1} e^{-\lambda_{i-1}\Delta t} \tag{5}$$

$$\lambda = f_{\mathbf{S}}(t) \tag{6}$$

where the ith EV arrives at time t_i, PDF represents the probability distribution function and t is time of day. The rate parameter λ is dependent on time, and $f_{\mathbf{S}}$ defines the profile of λ with respect to t for the type of days present in $\mathbf{S}$. We can use different methods to fit $f_{\mathbf{S}}$: The **mean model** is based on average values of λ for given timeslot t_s. This results in a discontinuous mapping between λ and t, with a sudden change in λ at the boundaries of each timeslot t_s. To have continuous λ throughout the day, we use regression-based methods: either a **polynomial model** using polynomial regression, or a **localized regression model**. Training these models is explained in detail in Section 4.1.1. In Algorithm 1, we outline the pseudocode to generate arrivals over a given horizon. We use the date (d) to retrieve the appropriate $f_{\mathbf{S}}$, and predict λ. The IAT between the current and new arrival is generated as a random sample from the exponential distribution with rate λ. Arrivals are generated throughout the horizon for each date.

Algorithm 1: Inter-arrival time (IAT) model.

Input : H (Horizon, initial to final date)
Output : T (List of EV arrival times in H)
for $d \in H$ **do**
  $f_{\mathbf{S}}$ = get arrival rate model for d;
  $t = 0$;
  while $t < 24$ **do**
    $\lambda = f_{\mathbf{S}}(t)$;
    Δt = sample from exponential distribution with rate λ;
    $t = t + \Delta t$;
    append t to list T;

2.1.2. Arrival Count Models

Instead of generating the next arrival of EV, here we focus on generating the number of arrivals in a given t_s (timeslot, e.g., slots of 60 min). The number of arrivals N in t_s can be generated as a random sample from a discrete probability distribution Equation (7). This distribution can be characterized using parameters $\mathbf{P}$, and Equation (6) can be modified to Equation (8), wherein we model these parameters. We distribute N arrivals uniformly over the duration of timeslot t_s. Arrival count (AC) models can be defined as follows:

$$PDF(N) = f(\mathbf{P}) \tag{7}$$

$$\mathbf{P} = f_{\mathbf{S}}(t_s) \tag{8}$$

We model the parameters $\mathbf{P}$ of the discrete distribution for each t_s using the function $f_{\mathbf{S}}$. Our underlying assumption that the IATs of EVs follow an exponential distribution amounts to assuming a Poisson distribution for the number of arrivals N in such a timeslot. Yet, for the Poisson distribution, the variance is equal to the mean of the distribution, while the number of arrivals may have a larger variance. In such case we need to include other discrete probability distributions that describe counts data [20]: we propose using the negative binomial model. In summary, we have two options to model the arrival counts (AC):

(1) **Poisson model:** Assuming that N follows a Poisson distribution (characterized by rate parameter λ; i.e., $\mathbf{P}$ is λ).
(2) **Negative binomial model:** Assuming that the N follows a negative binomial distribution ($\mathbf{P}$ is (μ, α)).

Pseudocode for generation of arrivals of EVs using the Poisson model is given in Algorithm 2 (adaptation to the negative binomial model for sampling N is straightforward).

Algorithm 2: Arrival count (AC) model.

```
Input  : H (Horizon, initial to final date)
Output : T (List of EV arrival times in H)
for d ∈ H do
    f_S = get arrival rate model for d;
    for t_s = 1, 2, ... 24 do
        λ = f_S(t_s);
        N = sample from Poisson distribution with rate λ;
        A = evenly space N points in t_s;
        append all t ∈ A to list T;
```

2.2. Mixture Models (MM_c, MM_e)

The connection time of each plugged-in EV depends on what time the EV arrived, i.e., its arrival time. We can model the probability distribution, $PDF_{t_a}(t_c)$ using gaussian mixture models (GMM), where t_c can be generated as a random sample from the probability distribution, Equation (9), once we know the value of t_a. We can group dates of a month (or daytype) into the same type of day, for which we use the same model. These dates then form a set $\mathbf{S}$ (set of dates). Similarly to the connected times, GMMs can be fitted for required energy (charging load).

$$MM_c : PDF_{t_a,\mathbf{S}}(t_c) \tag{9}$$

$$MM_e : PDF_{t_a,\mathbf{S}}(E) \tag{10}$$

The steps for data generation using SDG are summarized in Figure 1b. We used a trained SDG model and horizon as inputs. As seen in Figure 1a, we provided the methodology to train the models from a raw dataset. In Section 3 we describe the data cleaning and prepossessing, and session clustering steps. Then come the details of training and evaluation in Section 4.

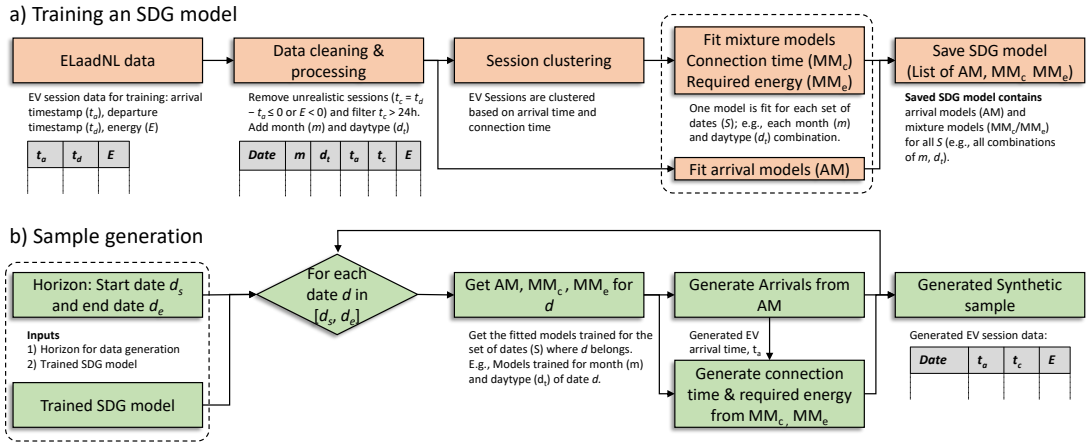

Figure 1. Modeling methodology for (**a**) training SDG models, and (**b**) generating synthetic samples.

In this section we define and outline the inputs of SDG, by defining AM for EV arrivals, and MM_c and MM_e for connection times and required energy. Inputs are simply the dates d (and arrival times t_a in case of MM_c and MM_e). We also summarize the parameters of SDG) by characterizing models using the parameters of the underlying probability distributions.

3. Dataset

The data used here were collected from ELaadNL (https://www.elaad.nl/), which is the knowledge and innovation center mutually associated with providers of charging infrastructure for the grid, to prepare for a future with electric mobility and sustainable charging. Operating since 2009, it has established a network of approximately 2000 public charging stations across The Netherlands. The EV session data collected by ELaadNL are not publicly available, and we obtained them based on an agreement. Furthermore, ElaadNL was not involved in the study, and acted only as a data provider. People interested in the dataset are encouraged to contact us. In this section, we provide the details of the data cleaning and processing, and session clustering steps (Figure 1a).

3.1. SDG Training Data

The EV sessions' time series data were prepared for training the SDG (the training process is detailed in Section 4). These data contain: the date d, month m, type of day d_t, arrival time $tarr$, arrival timeslot t_s, connection time t_c and required energy E, as shown in Table 1.

Timeslots have values ranging from 1 to 24, where 1 indicates the timespan 00:00–00:59, 2 indicates 01:00–01:59, etc. Further, t_a and t_c are real numbers ($\in [0, 24)$); e.g., 1.5 means 01:30 A.M. More than 98% of the sessions have t_c under 24 h, so we safely assumed that the maximum connection time was 24 h (and removed data points with $t_c > 24$). In the real world, we will have sessions where the EV departs before it is fully charged. However, the collected data do not include the charging load that was unmet before the EV departed. Lacking such information, we resorted to assuming the measured charging load represents fully charging the EV. We represent this charging load, or energy required by E in kWh.

Further, the training data were properly cleaned, which included removing impractical or incorrect sessions parameters (where $E < 0$ or $t_a = t_d$).

Table 1. Processed session data. Each row corresponds to an EV session.

d Date	m Month	d_t Day Type	t_a Arrival Time (h)	t_s Arrival Time Slot	t_c Connection Time (h)	E Required Energy (kWh)
01/01/2015	1	0	0.15	1	4.3	3
…	…	…	…	…	…	…

3.2. Charging Stations Analysis

The full ELaadNL dataset contains 1.8 million sessions from January 2012 till June 2018. The infrastructure consists of charging stations of 10 different types, divided by manufacturer type, charging speed and other factors. In 2016 the EVnetNL (the infrastructure provider associated with ELaadNL) stations were transformed to integrate smart charging capability. Hardware and software of the charging stations (poles) were updated based on the station type. In 2017, more than 50% of EVnetNL stations were taken over by other charging station operators. Due to those two factors, we observed a sudden drop in the number of daily active charging stations in 2016 and 2017 (Figure 2). The years prior to 2014 have a very steep growth curve in terms of active poles, while from 2016 onwards, the active poles become unpredictable because of market factors. As we wanted our model to reflect charging behavior, and not be influenced by infrastructure changes, we selected the training data from the reasonably stable year 2015. The data used for training our SDG were from January to December 2015 of the ElaadNL dataset. This data contains 365,000 sessions. In 2015, the number of used poles amounted to 1677, out of which 1645 poles were active before and after 2015. We used the data from these 1645 poles for our analysis. Thus, we considered a constant number of poles to construct our SDG model, and avoided the effects of a changing number of EV charging stations.

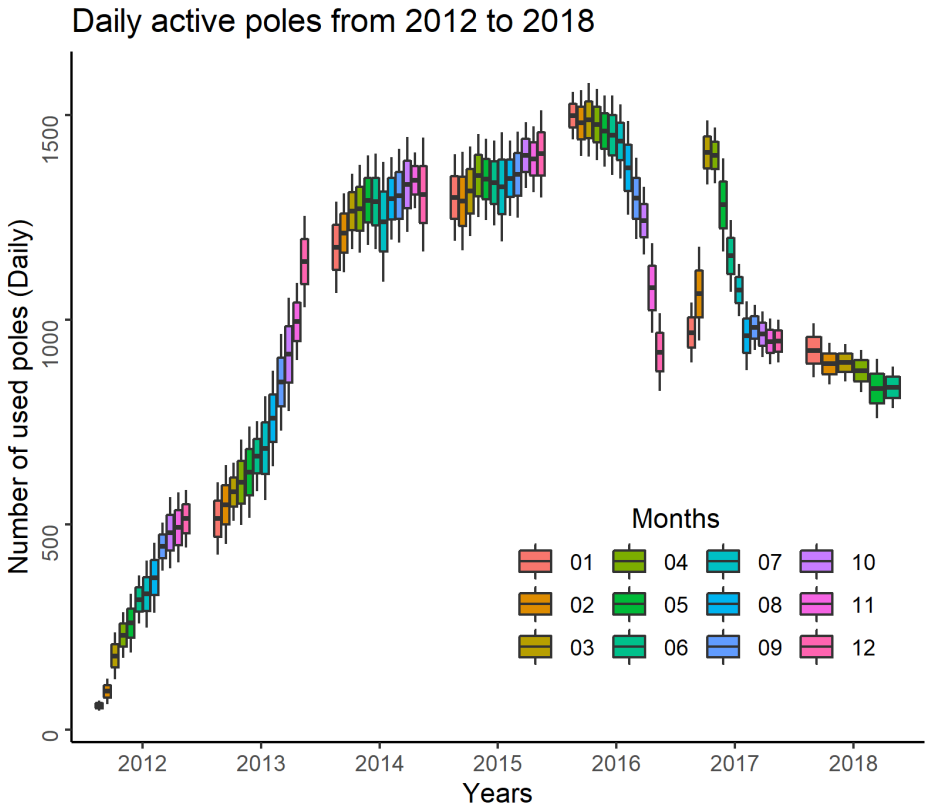

Figure 2. Number of used poles per day, from 2012 to 2018. Each boxplot represents data for 1 month. The *y*-axis represents the number of daily active poles.

3.3. Clustering

We used the expectation maximization (EM) algorithm for training the GMM used in MM_c (defined in Section 2.2). EM algorithm can be initialized with a realistic number of mixtures (along with mean and variance for each mixture) for it to converge to a practical solution. To achieve this practical solution, we initialized each GMM with session clusters. EV sessions were clustered based on arrival and connection times, and here we outline the different types of sessions that are observed in the real-world data.

Sessions clusters: In our previous work [2], on the same data, we discussed three types of sessions. Namely, (i) **Park to charge**: arrivals throughout the day; (ii) **Charge near home**: arrivals during evenings, and staying till late at night; (iii) **Charge near work**: arrivals during early morning, and staying till evenings. The largest cluster was the park to charge cluster (60% of sessions), followed by the charge near home (29% of sessions) and the charge near work clusters (11% of sessions). The DBSCAN algorithm was used to determine these clusters, which is a density based clustering algorithm. We could see a similar distribution of sessions in the 2015 dataset, after clustering the sessions. The resulting session clusters are shown in Figure 3. These clusters are only based on 2015 data, contrary to the previous work, which combined the full data for 2012–2016. Please refer to [2] for further details.

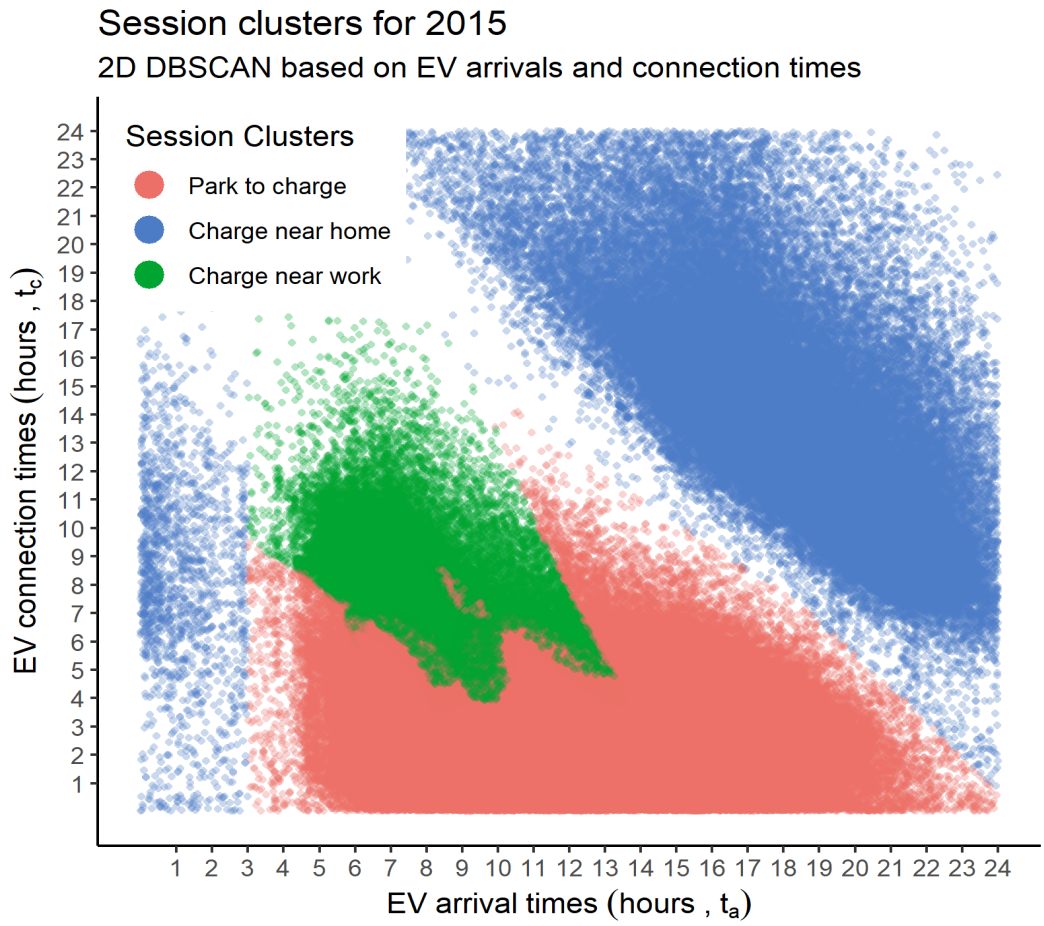

Figure 3. Session clusters for 2015. We used DBScan to cluster EV sessions on a monthly basis, and combine the data for all months.

4. Training Additionally, Evaluation

4.1. Training

We used the training data of 2015, as described in Section 3.1, for training and evaluating our models. Aggregating EV sessions on a monthly basis reveals a higher number of sessions during the winter months, compared to the summer months. In the case of daily EV sessions, it has been noticed that all weekdays have similar profiles, which are different from weekends [12]. Thus, we could assume that days belonging to the same month (m) and daytype (d_t, weekday vs. weekend) have similar profiles, and defined a set of dates (**S**) as pairs (m, d_t) (e.g., for m = January and d_t = weekday, **S** will have all dates that are weekdays from January). Training data for the model for a (m, d_t) combination comprises the session data for dates present in the respective **S**. We trained 24 individual models, one for each of the (m, d_t) combination.

4.1.1. Arrival Model

Inter-arrival time models: For IAT models, we modeled the daily profiles of λ, which can be fitted using (i) a mean model, (ii) a polynomial regression model or (iii) a localized regression model (outlined in Section 2.1.1). We can rewrite Equation (6) in terms of (m, d_t) as in Equation (11). We calculated the EV arrival rates (λ) for each day and t_s (24 timeslots of 60 min each), by fitting the inter-arrival time to an exponential distribution.

$$\lambda_{m,d_t} = f_{m,d_t}(t) \tag{11}$$

For the mean model, the fitted value for each t_s is the average λ (across all days). Accordingly, each t_s has a single value of λ. This results in a discontinuous mapping between λ and t, for which the function in Equation (11) becomes discontinuous at the boundaries of t_s and we see a sudden change in λ.

For regression methods, we transform λ by taking the logarithm and applying min-max normalization for each day. This transformation is necessary to correctly fit the peak hours, during which inter-arrival times are very low (high λ). We take the logarithm in order to more accurately model the values of λ during the night hours (00:00–06:00), which have few arrivals (low λ). Normalization is used to scale the arrival rates of all days in **S** to the same levels. We represent this transformed λ using λ_t, and we use s to represent the re-scaling parameter for predicted values. Equation (13) can be used to get the fitted λ from the regression models $f_{m,d_t}(t)$.

$$(\lambda_t)_{m,d_t} = f_{m,d_t}(t), \quad 0 < \lambda_t \leq 1 \tag{12}$$

$$\lambda_{m,d_t} = e^{s f_{m,d_t}(t)} \tag{13}$$

For the polynomial regression model we modeled the relationship in Equation (12) using a grid search for the best polynomial degree ($\in \{1, \ldots, 50\}$). Mean squared error (MSE) was used as the error metric during grid searching. It provides a strong penalty for large errors, which was necessary to fit the sharp morning peaks during weekdays. This resulted in a continuous and differentiable function of λ in terms of t.

For the localized regression model, polynomials of degree 1 and 2 with $\alpha \in \{0.125, 0.25, 0.5\}$ were tested. We noticed that the best results sdfd generated for degree 1 and $\alpha = 0.125$. This resulted in a piecewise, continuous and differentiable profile of λ throughout the day.

Scale treatment and randomization: For regression based methods, for which we model λ_t, we may encounter a situation wherein $f(t) < 0$. In this case, λ becomes very low, which can cause the sampled inter-arrival time (Δt) of EVs to be very large. Since the next EV arrival is calculated relative to the past arrival Equation (4), such high Δt may cause the next arrival to be very late, thereby skipping a large period of time. This becomes problematic when this period covers times with high

values of λ (hence a high number of EV arrivals, which however will not be generated). For practical purposes, we impose a lower limit of 1 on λ (meaning we have at least 1 arrival in each t_s).

When we transformed λ, we appiedy a min-max normalization on $\ln(\lambda)$ for each day (where the minimum value of $\ln(\lambda)$ is 0, because $\lambda \geq 1$). Each day in the training data has its own maximum value of $\ln(\lambda)$. These values can be saved as an array of re-scaling parameters (represented by s in Equation (13)). When generating session arrivals, we randomly selected a value from this array to re-scale the predicted values. This helped in introducing variance in to the otherwise smooth profiles of the predicted λ.

Arrival count models: We mapped each t_s to the parameters $\mathbf{P}$ that characterize the discrete distribution of the number of arrivals in that timeslot. Similarly to the IAT models, we have a model for each (m, d_t) combination. Our training data for each model are the numbers of EV arrivals at t_s for each of the days of the respective combination (m, d_t).

$$\begin{array}{rccc} f_{m,d_t}: & t_s & \rightarrow & \mathbf{P}_{m,d_t} \\ & \{1,\ldots,24\} & \rightarrow & \{\mathbf{P}_{m,d_t,1},\ldots,\mathbf{P}_{m,d_t,24}\} \end{array} \tag{14}$$

For the Poisson model, the average number of EV arrivals λ was calculated per t_s. In the Poisson distribution, the mean is equal to its variance, a restriction that is not present in the negative binomial distribution.

In case of the negative binomial model (with parameters $\mathbf{P} = \{\mu, \alpha\}$), μ is the average number of EV arrivals per timeslot (=λ), and α is the dispersion parameter, which can be used to define the variance of the distribution (var $= \mu + \alpha\mu^2$). A negative binomial distribution model thus allows one to introduce more variability in the generated number of EV arrivals, compared to a Poisson model. Both α and μ were fitted for each individual (m, d_t) combination. It is possible that the estimated α for an (m, d_t) combination is extremely low, in which case the underlying distribution is more likely to be Poisson. It is also possible that during night hours (low EV arrivals), the estimation process of α might result in impractical values (less than 0). To adjust for this, we can use a Poisson distribution wherein the estimated values of α are negative, or set a lower limit on α (e.g., $\alpha \geq 0.1$).

In IAT models, the time of the next EV arrival is the sum of the time of previous EV arrival and randomly sampled Δt. As previously stated, this dependency becomes troublesome if Δt is very large (due to the low λ, the next EV arrival may be very late, skipping a large time interval) or very low (high λ, large number of EV arrivals in a small amount of time). Due to that, fitting λ as a function of t requires caution in IAT models. However, we do not face this problem when using the AC modeling approach, wherein the number of arrivals are generated separately for each t_s. Indeed, a low/high λ in the previous t_s will not affect the number of arrivals in next t_s. For practical uses, we can also assume that night hours with low numbers of EV arrivals are similar, and combine t_s = 1–6 into a single timeslot. The fitted value of λ is then associated with the time from 00:00 to 06:00.

4.1.2. Mixture Models (MM_c, MM_e)

Similarly to the arrival models, we verified that days belonging to a (month, daytype) combination have similar distributions in terms of departure times and charging loads, and thus fit models for each (m, d_t) combination. For each t_s we fit a Gaussian mixture model to the real-world data, and modified Equation (9) as follows.

$$\begin{aligned} P_{t_a=t_s,m,d_t}(t_c) &= GMM_{m,d_t,t_s} \\ &= \left(\textstyle\sum_{k=1}^{K} \phi_k \mathcal{N}(\mu_k, \sigma_k^2)\right)_{m,d_t,t_s} \end{aligned} \tag{15}$$

This resulted in a GMM fitted for each (m, d_t, t_s) combination, with trained values for (i) mixing probabilities (ϕ_k), (ii) mixture means (μ_k) and (iii) mixture variances (σ_k^2), for each mixture. We used expectation minimization to fit the GMM.

Expectation maximization for fitting GMM requires initialization of mixtures (μ_k, σ_k^2). The number of mixtures also needs to be chosen for each model (representing a m, d_t, t_s combination). We initialized each GMM based on session clusters (see Section 3.3). We grouped the sessions observed in m, d_t, t_s, into their respective session clusters. We used the number of clusters obtained to initialize the K for the GMM, and calculated the μ_k, σ_k^2 from the EV sessions in the respective groups.

4.2. Evaluation

Exponential distribution: We performed a Kolmogorov–Smirnov (KS) goodness-of-fit test to validate the assumption that inter-arrival times of EV sessions follow the exponential distribution.

Arrival models: Once the AM was trained using the 2015 EV session data, a synthetic sample for 2015 could be generated. This sample was to generate EV arrivals from January 1, 2015 to December 31, 2015. We generate 10 samples for each modeling method (three IAT models and two AC models). EV arrivals were aggregated on an hourly and daily basis. Since the aggregated values represent count data, we used a non-parametric Wilcoxon test to assess similarity between the generated samples and the actual data. We performed the test on a monthly basis for the daily aggregated data and on an hourly basis for hourly aggregated data. We provide plots for visual comparison.

Mixture models: Connection times were sampled from the fitted GMMs, for the actual EV arrivals. Density plots were created to evaluate whether the peaks of the conditional probability distributions were modeled correctly. A similar evaluation was preformed for required energy.

SDG: Final generated data (and actual data) were 3-dimensional, with each session defined by (t_a, t_c, E). The actual data comprised 350,000 sessions, and the numbers of sessions in the generated samples were of the same order. Since two-sample similarity tests for high dimensional data become unreliable as the data size increases, we used a kernel density estimation (KDE) test [21] and a multidimensional version of the KS test [22,23]. We did those tests for (t_a, t_c) and (t_a, E) combinations.

In this section we defined different methods for fitting the parameters of SDG. Depending on the modeling method, the parameters of SDG will also change (λ in case of the exponential distribution, and (μ , α) in case of the negative binomial distribution).

5. Results

5.1. Assumptions

KS test p-values are greater than 0.05 for each hour of the day, as plotted in Figure 4. This validates that the inter-arrival times of EV sessions are exponentially distributed (Section 2.1), and thus supports our chosen models AM of the arrival times.

5.2. Distribution of Arrival Rates λ

To understand how the SDG parameters change with inputs, we have plotted the profiles of λ for weekend and weekdays for 2015 in Figure 5. We see a similar pattern for all months. Arrival models were fitted to approximate this behavior of λ. On weekdays, we see two peaks in the profile of λ that represent high frequencies of EV arrivals.

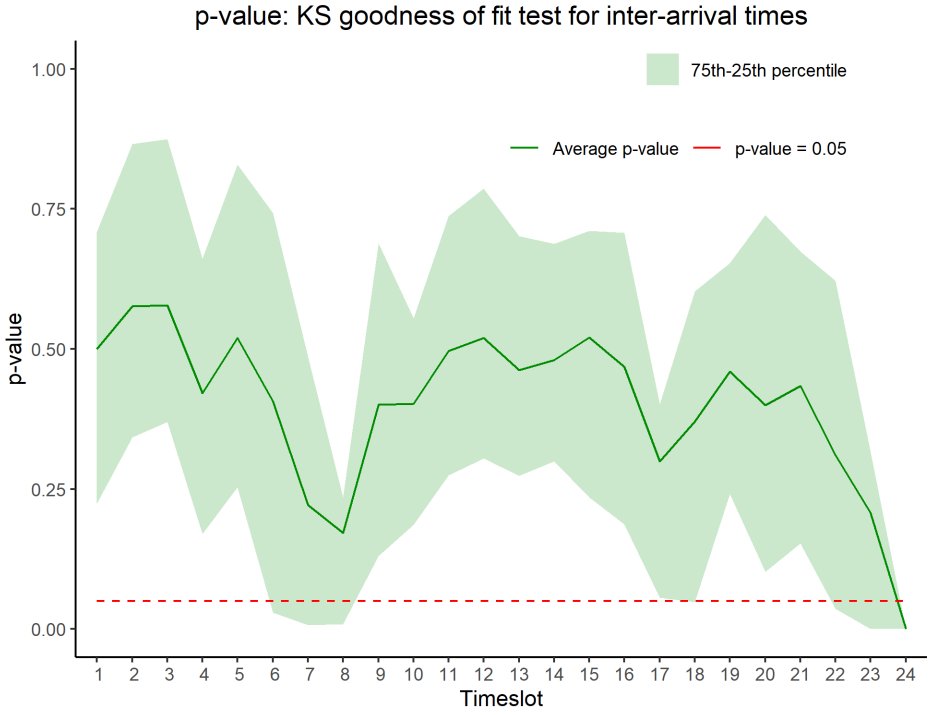

Figure 4. KS test p-values: For each (m, d_t) combination, 24 KS tests were performed for each timeslot (t_s). High p-values indicate that null hypotheses (IATs are exponentially distributed) could not be rejected.

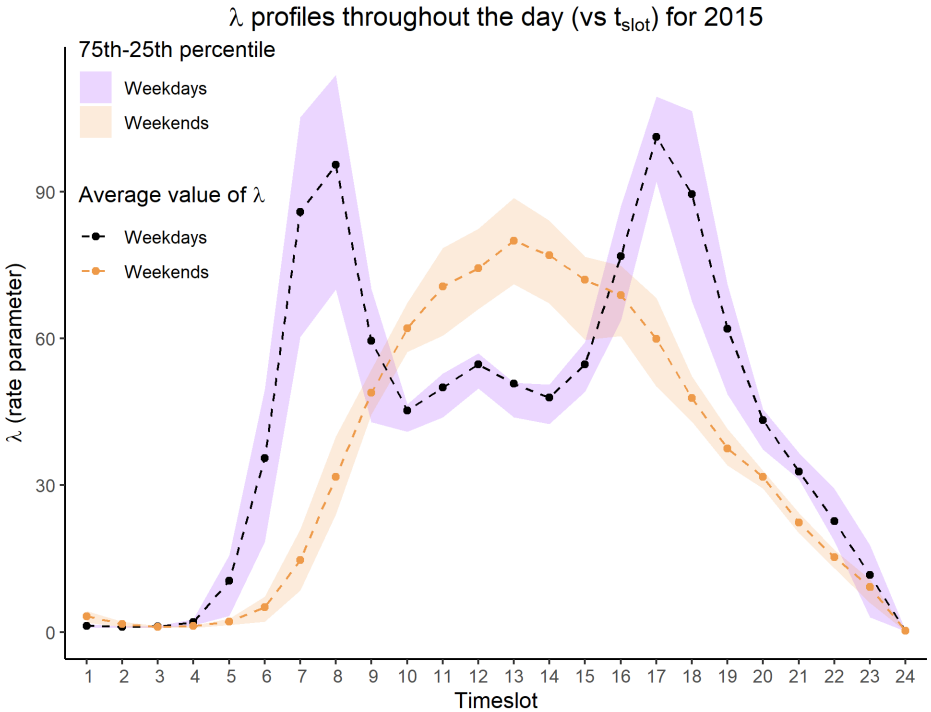

Figure 5. Daily λ profiles for 2015: For each (m, d_t) combination, we calculated the average arrival rate λ. The dotted line represents the average over those 12 months; the shaded areas indicate the percentile range.

5.3. Arrival Models (AM)

We generated 10 samples of arrivals of EVs for 2015 for both inter-arrival time (IAT) and arrival count (AC) models. The total number of arrivals per day was calculated and plotted in Figure 6. Similarly, Figure 7 shows the aggregated hourly EV arrivals. Both these plots are for weekdays, and similar results were observed in case of weekends. We can clearly see that the generated data are very similar to the actual data. We further quantitatively compared the values of the actual EV arrivals with the generated EV arrivals using a Wilcoxon test. The null hypothesis was that the means of these are equal. High p-values (> 0.05) indicate that the daily generated arrivals are statistically similar to the actual data. This is represented by *ns* in the figure, implying that the difference between the two samples is not significant. The results presented are for comparisons between one month of actual data to 10 samples of the same month of generated data. We got similar results when we compare the real-world data to a single sample.

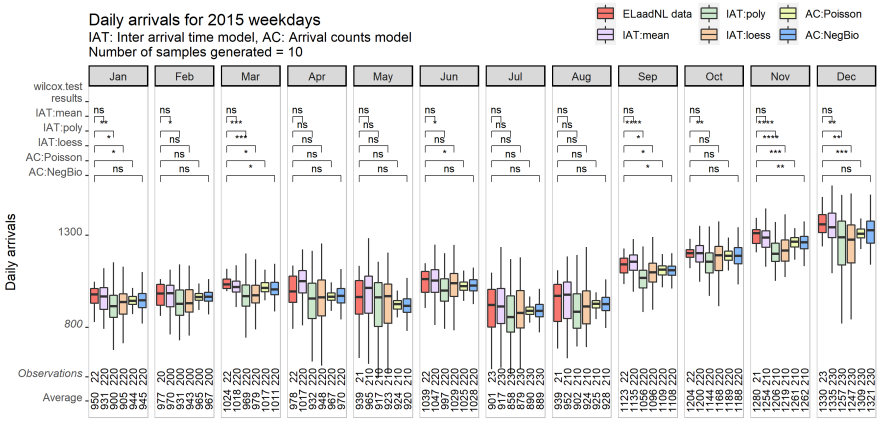

Figure 6. Daily aggregated EV arrivals (2015, weekdays). Significance was calculated based on Wilcoxon tests (ns: not significant, p-value > 0.05; * p-value $\leq$ 0.05; ** p-value $\leq$ 0.01; *** p-value $\leq$ 0.001; **** p-value $\leq$ 0.0001).

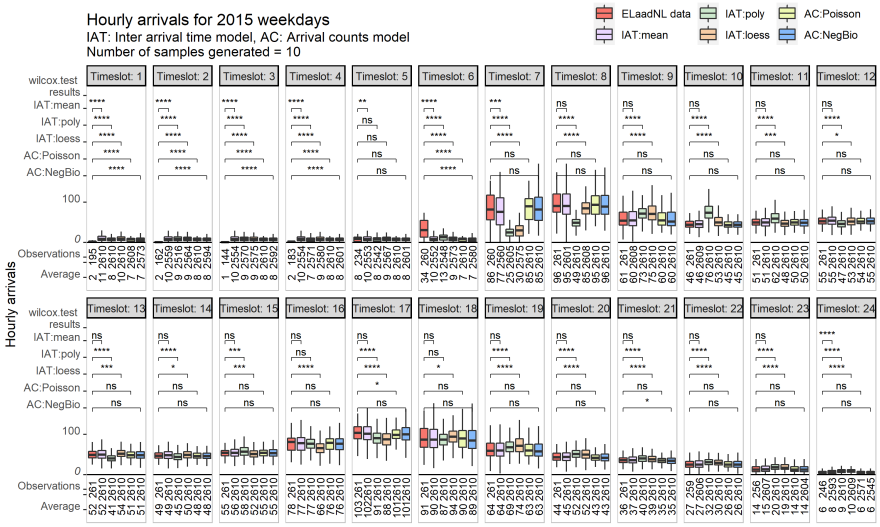

Figure 7. Hourly aggregated EV arrivals (2015, weekdays). Significance was calculated based on Wilcoxon tests (ns: not significant, p-value > 0.05; * p-value $\leq$ 0.05; ** p-value $\leq$ 0.01; *** p-value $\leq$ 0.001; **** p-value $\leq$ 0.0001).

5.4. Mixture Models (MM_c, MM_e)

Conditional distributions for connection times (hours) and energy required (kWh) are plotted in Figures 8 and 9 respectively. The plots on the left were created from the real-world data, and those on the right were created from the data generated from mixture models (MM_c, MM_e). These figures are for weekdays, and similar plots were generated for weekends. Connection times (and energy required) were generated using GMM and real-world EV arrivals. Vertical divisions in the generated data for each times slot can be seen, because we use one GMM per $\{m, dt, t_s\}$ combination.

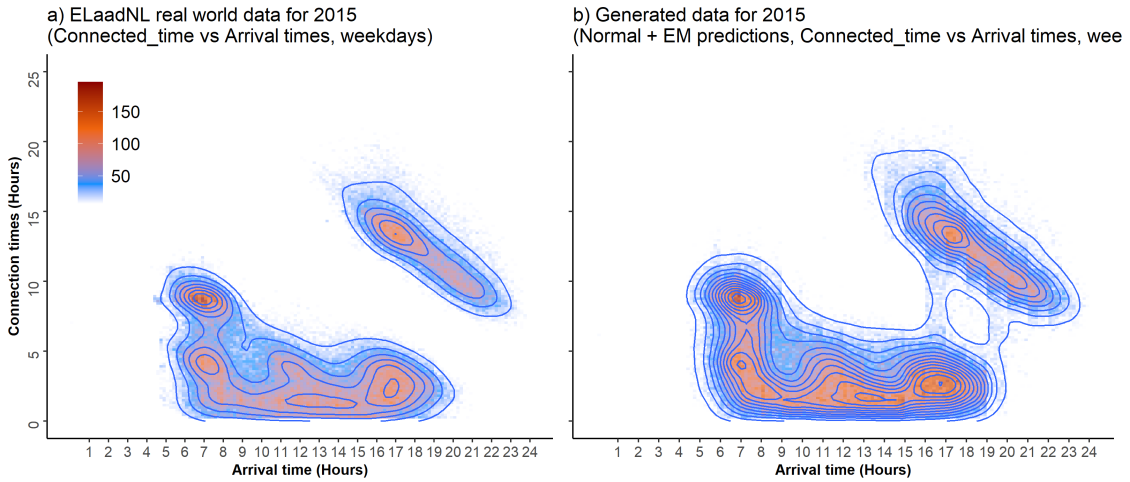

Figure 8. Density plots for connection times (2015, weekdays). Generated data represent sampled connection times for real-world EV arrivals. Each point represents a bin (10 min by 10 min), and is colored based on the number of EV sessions in the bin (bins with less than 5 arrivals were not plotted to keep the graph readable).

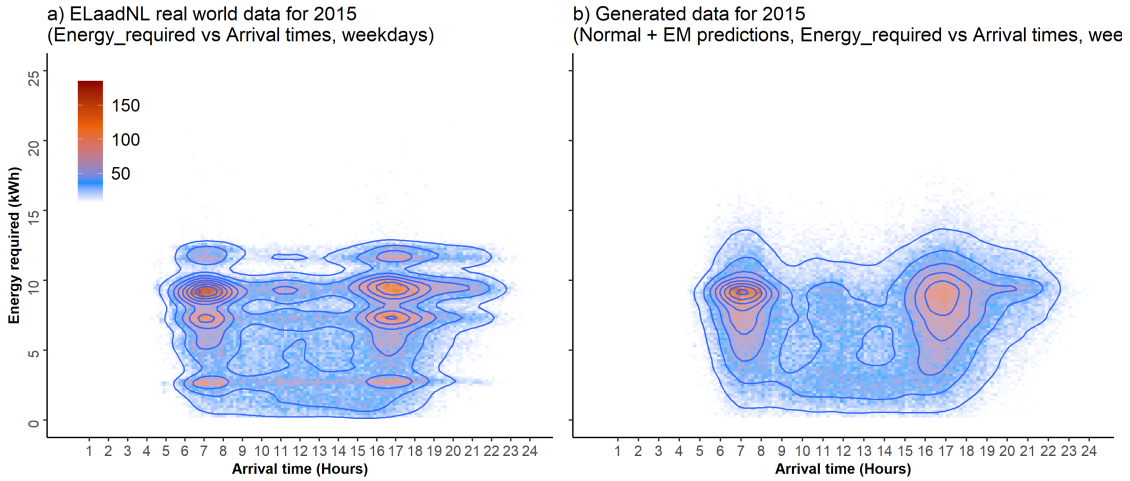

Figure 9. Density plots for required energy (2015, weekdays). Generated data represent sampled energy requirements for real-world EV arrivals. Each point represents a bin (10 min by 0.16 kWh), and is colored based the number of EV sessions in the bin (bins with less than 5 arrivals were not plotted to keep the graph readable).

5.5. Synthetic Data Generator (SDG)

We generated full session samples including generated arrival times, connection times and required energy for all the models. These generated samples were compared with the real-world session data for 2015. To compare the models' synthetic samples with real-world data, 2-sample KDE

tests were performed. In Figure 10a, we show the KDE test for (t_a, t_c), and Figure 10b shows the KDE test results on (t_a, E). As we can see, the mean model for the IAT, and both models for AC have average p-values > 0.05, indicating that the generated data are similar to the real-world EV session data. We observed similar results in the multidimensional KS test. These results conclude that the generated samples are statistically similar to real-world data.

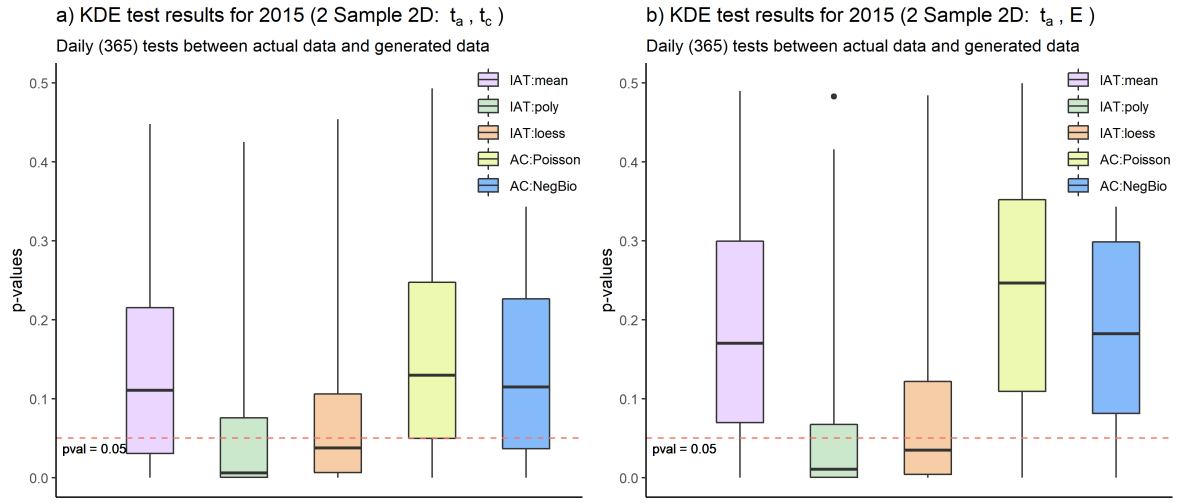

Figure 10. KDE test p-values: Daily 2 sample 2 dimensional KDE tests to compare real-world and generated data. Total of 365 tests performed for each model. p-value > 0.05 means datasets are similar. (**a**) Results for (arrival times t_a, connection times t_c). (**b**) Results for (arrival times t_a, energy required E).

6. Discussion

In this paper, we proposed a synthetic data generator (SDG) to create samples of realistic EV session data. Each session is defined by arrival time, departure time and required energy. We described two modeling methodologies to generate arrivals, assuming that inter-arrival times follow exponential distribution. Different methods for modeling the daily profiles of the parameter λ were tested. For connection times and required energy, mixture models were trained to estimate the probability distributions. Our real-world dataset was used to train the SDG, and multiple samples of session data were generated.

Inter-arrival times followed an exponential distribution, which was validated by KS test results. Wilcoxon tests were used to compare daily and hourly EV arrivals from the generated samples and real-world data (Figure 4).

Arrival count (AC) models performed better compared to inter-arrival time (IAT) models. The negative binomial model from the AC models outperformed all the other models for generating EV arrivals. Samples generated by the IAT model exhibited high variance, which was introduced during the scale treatment and randomization step (Section 4.1.1). In the IAT models, we note that regression methods failed to capture the morning peak in the hourly arrivals. This occurred because the regression curves failed to capture the sharp increase in the number of arrivals. We indeed see that the polynomial model (IAT:poly) and localized regression model (IAT:loess) generated very low numbers of samples during morning peaks ($t_s = 7, 8$). In contrast, the AC models were able to capture both morning and evening peaks, as can be seen in Figure 7. The AC models also captured the variance in number of arrivals throughout all t_s. During night hours, we noticed a difference between the generated and real-world data. However, for practical purposes, this difference can be neglected as the average number of arrivals is very low. We can see that the negative binomial model performs best for both daily and hourly generation. This makes it ideal for both short and long-term data generation.

In our mixture models, Gaussian mixture models (GMM) were able to properly capture peaks of the conditional probability distributions. We clustered EV sessions based on arrival times and

connection times, and each peak in the conditional probability distribution corresponds to one session cluster. Two peaks in required energy distribution represent the morning and evening demand. For the connection times, we can see that after generating the data, all three session cluster peaks were captured (Figure 8). In case of required energy, both morning and evening peaks were captured (Figure 9). Since we were able to capture the probability distributions of t_c and E, GMM-based mixture models could be used for fitting the conditional distributions.

In case of weekdays, during some spring and summer months we observed a very high variance in the number of daily arrivals in actual data (in Figure 6). The reason is that there are multiple holidays during May, July and August that have very low numbers of arrivals. In retrospect, we found that many of these holidays (on weekdays) have arrival profiles similar to weekends. Due to that, arrival models trained for weekdays are unable to capture this variance. Introducing holidays as another daytype (d_t) or modeling holidays as weekends should help to overcome this limitation.

We modeled the data under the assumption that the number of active charging stations would remain constant during the time in which the EV session data are collected. Furthermore, the generated sample is representative of EV sessions that might occur on this constant number of active charging stations. Hence, the proposed methodology does not model the effect of changing the number of charging stations, where future research is possible.

7. Conclusions

EV session data collected from charging stations on a electricity gird can be used for flexibility analysis, making pricing decisions, etc., and are essential for advancement in the field of smart grids.

We defined a synthetic data generator (SDG) to generate samples of EV session data collected on charging stations. We modeled arrival times of EVs using inter-arrival time (IAT) and arrival counts (AC) methods. For generating the connection times and required energy, we used mixture models based on GMM. The generated sample of session data is statistically indistinguishable from the real-world data, as seen from the KDE test results. We can conclude that our proposed SDG is suited for generating a synthetic sample of EV session data.

This generated data sample will have the properties of a real-world EV sessions, and can be used for purposes such as flexibility analysis. We will release the trained SDG models that can be used to generate new samples of EV session data. Complete code for training and evaluating the SDG models is open source, and can be used to fit the models on a new EV session data (see Appendix A). These models can be shared without violating the privacy concerns of the real data collection companies.

For future work, further exploration is required in studying reduced variance in daily arrivals in AC models. IAT models for arrival times misses the first peak of weekdays, wherein improvements are possible. A deeper dive into the mixture models for estimating the conditional distribution of required energy can also provide an improvement to results.

Author Contributions: M.L. developed the methodology and performed data curation, validation, visualization, software and writing of the original draft; C.D. handled resources and funding acquisition; D.F.B. and C.D. jointly supervised the work presented, and performed review and editing of the paper. All authors have read and agreed to the published version of the manuscript.

Funding: This research received funding from the Flemish Government under the "Onderzoeksprogramma Artificiële Intelligentie (AI) Vlaanderen" programme.

Acknowledgments: We thank Nazir Refa from ElaadNL for the real-world EV session data.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Code

Code for training SDG models is open source, and can be accessed on GitHub: https://github.com/mlahariya/EV-SDG. SDG models trained with a real-world dataset are also included with the code. These can be used to generate a random sample of EV session data using script

SDG_sample_generate.py. Trained models that can be used as default models to generate samples with are located at modeling/default_models, and include:

- ***SDG Model (IAT,mean)***: IAT model based on mean model.
- ***SDG Model (IAT,poly)***: IAT model based on polynomial regression model.
- ***SDG Model (IAT,loess)***: IAT model based on localized regression model.
- ***SDG Model (AC,poisson_fit)***: AC model based on Poisson distribution.
- ***SDG Model (AC,neg_bio_reg)***: AC model based on negative binomial distribution.

Users can also employ our code to fit AM, MM_c, and MM_e to their own datasets. For training a SDG model from scratch, this process will be followed: (i) Clean real-world EV session data (***preprocess***). (ii) Generate session and pole clusters (***preprocess***). (iii) Prepare data for SDG training (***preprocess***). (iv) Train AM, MM_c and MM_e models (***modeling***). (v) Save the model along with a log file in the 'res/' folder. A command line callable script SDG_fit.py can be used to fit the models.

Please visit the repository for further details.

References

1. Watson, R.T.; Boudreau, M.C.; Chen, A.J. Information systems and environmentally sustainable development: Energy informatics and new directions for the IS community. *MIS Q.* **2010**, *34*, 23–38. [CrossRef]
2. Develder, C.; Sadeghianpourhamami, N.; Strobbe, M.; Refa, N. Quantifying flexibility in EV charging as DR potential: Analysis of two real-world data sets. In Proceedings of the 2016 IEEE International Conference on Smart Grid Communications (SmartGridComm), Sydney, Australia, 6–9 November 2016; pp. 600–605.
3. Quirós-Tortós, J.; Espinosa, A.N.; Ochoa, L.F.; Butler, T. Statistical representation of EV charging: Real data analysis and applications. In Proceedings of the 2018 Power Systems Computation Conference (PSCC), Dublin, Ireland, 11–15 June 2018; pp. 1–7.
4. Islam, E.; Moawad, A.; Kim, N.; Rousseau, A. *An Extensive Study on Sizing, Energy Consumption, and Cost of Advanced Vehicle Technologies*; Argonne National Lab. (ANL): Argonne, IL, USA, 2018.
5. Hanke, C.; Hüelsmann, M.; Fornahl, D. Socio-Economic Aspects of Electric Vehicles: A Literature Review. In *Evolutionary Paths Towards The Mobility of the Future*; Hüelsmann, M., Fornahl, D., Eds.; Springer: Berlin/Heidelberg, Germany, 2014; pp. 13–36. [CrossRef]
6. Li, X.; Chen, P.; Wang, X. Impacts of renewables and socioeconomic factors on electric vehicle demands: Panel data studies across 14 countries. *Energy Policy* **2017**, *109*, 473–478. [CrossRef]
7. Pevec, D.; Babic, J.; Podobnik, V. Electric vehicles: A data science perspective review. *Electronics* **2019**, *8*, 1190. [CrossRef]
8. Sadeghianpourhamami, N.; Deleu, J.; Develder, C. Definition and evaluation of model-free coordination of electrical vehicle charging with reinforcement learning. *IEEE Trans. Smart Grid* **2019**, *11*, 203–214. [CrossRef]
9. Mies, J.; Helmus, J.; van den Hoed, R. Estimating the charging profile of individual charge sessions of electric vehicles in the Netherlands. *World Electr. Veh. J.* **2018**, *9*, 17. [CrossRef]
10. Zhang, C.; Kuppannagari, S.R.; Kannan, R.; Prasanna, V.K. Generative adversarial network for synthetic time series data generation in smart grids. In Proceedings of the 2018 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm), Aalborg, Denmark, 29–31 December 2018.
11. Shepero, M.; Munkhammar, J. Data from Electric Vehicle Charging Stations: Analysis and Model Development. In Proceedings of the 1st E-Mobility Power System Integration Symposium, Berlin, Germany, 23 October 2017.
12. Flammini, M.G.; Prettico, G.; Julea, A.; Fulli, G.; Mazza, A.; Chicco, G. Statistical characterisation of the real transaction data gathered from electric vehicle charging stations. *Electr. Power Syst. Res.* **2019**, *166*, 136–150. [CrossRef]
13. Brady, J.; O'Mahony, M. Modelling charging profiles of electric vehicles based on real-world electric vehicle charging data. *Sustain. Cities Soc.* **2016**, *26*. [CrossRef]
14. Gadda, S.; Kockelman, K.M.; Damien, P. Continuous departure time models: A bayesian approach. *Transp. Res. Rec.* **2009**, *2132*, 13–24. [CrossRef]

15. Sadeghianpourhamami, N.; Benoit, D.; Deschrijver, D.; Develder, C. Bayesian cylindrical data modeling using Abe-Ley mixtures. *Appl. Math. Model.* **2018**, *68*, 629–642. [CrossRef]
16. Sadeghianpourhamami, N.; Benoit, D.; Deschrijver, D.; Develder, C. Modeling real-world flexibility of residential power consumption: Exploring the cylindrical WeiSSVM distribution. In Proceedings of the Ninth International Conference on Future Energy Systems, Karlsruhe, Germany, 12–15 June 2018; pp. 408–410.
17. Majidpour, M.; Qiu, C.; Chu, P.; Gadh, R.; Pota, H.R. Fast prediction for sparse time series: Demand forecast of EV charging stations for cell phone applications. *IEEE Trans. Ind. Informatics* **2015**, *11*, 242–250. [CrossRef]
18. Iftikhar, N.; Liu, X.; Danalachi, S. A scalable smart meter data generator using spark. In Proceedings of the OTM Confederated International Conferences On the Move to Meaningful Internet Systems, Rhodes, Greece, 23–28 October 2017.
19. Lahariya, M.; Benoit, D.; Develder, C. Defining a synthetic data generator for realistic electric vehicle charging sessions. In *Proceedings of the Eleventh ACM International Conference on Future Energy Systems*; Association for Computing Machinery: New York, NY, USA, 2020; p. 406–407. [CrossRef]
20. Cameron, A.; Trivedi, P. Count Panel Data. In *The Oxford Handbook of Panel Data*; Baltagi, B.H., Ed.; Oxford University Press: Oxford, UK, 2015. [CrossRef]
21. Duong, T.; Goud, B.; Schauer, K. Closed-form density-based framework for automatic detection of cellular morphology changes. *Proc. Natl. Acad. Sci. USA* **2012**, *109*, 8382–8387. [CrossRef] [PubMed]
22. Fasano, G.; Franceschini, A. A multidimensional version of the Kolmogorov–Smirnov test. *Mon. Not. R. Astron. Soc.* **1987**, *225*, 155–170. [CrossRef]
23. Peacock, J.A. Two-dimensional goodness-of-fit testing in astronomy. *Mon. Not. R. Astron. Soc.* **1983**, *202*, 615–627. [CrossRef]