Accepted for publication in IEEE SmartgridComm 2018

Generative Adversarial Network for Synthetic Time Series Data Generation in Smart Grids

Chi Zhang*, Sanmukh R. Kuppannagari†, Rajgopal Kannan† and Viktor K. Prasanna†
*Computer Science Department, University of Southern California, Los Angeles, CA
†Electrical Engineering Department, University of Southern California, Los Angeles, CA
Email: zhan527@usc.edu, kuppanna@usc.edu, rajgopak@usc.edu, prasanna@usc.edu

***Abstract*—The availability of fine grained time series data is a pre-requisite for research in smart-grids. While data for transmission systems is relatively easily obtainable, issues related to data collection, security and privacy hinder the widespread public availability/accessibility of such datasets at the distribution system level. This has prevented the larger research community from effectively applying sophisticated machine learning algorithms to significantly improve the distribution-level accuracy of predictions and increase the efficiency of grid operations.**

Synthetic dataset generation has proven to be a promising solution for addressing data availability issues in various domains such as computer vision, natural language processing and medicine. However, its exploration in the smart grid context remains unsatisfactory. Previous works have tried to generate synthetic datasets by modeling the underlying system dynamics: an approach which is difficult, time consuming, error prone and oftentimes infeasible in many problems. In this work, we propose a novel data-driven approach to synthetic dataset generation by utilizing deep generative adversarial networks (GAN) to learn the conditional probability distribution of essential features in the real dataset and generate samples based on the learned distribution. To evaluate our synthetically generated dataset, we measure the maximum mean discrepancy (MMD) between real and synthetic datasets as probability distributions, and show that their sampling distance converges. To further validate our synthetic dataset, we perform common smart grid tasks such as k-means clustering and short-term prediction on both datasets. Experimental results show the efficacy of our synthetic dataset approach: the real and synthetic datasets are indistinguishable by solely examining the output of these tasks.

I. Motivation

The lack of fine grained distribution system data is a significant bottleneck preventing the community from developing novel data science and machine solutions for smart grid applications such as load forecasting [1], dynamic demand response [2], behind-the-meter disaggregation [3] and so on. Although efforts exist to make public datasets available [4][5], they are limited due to the following reasons:

- **Availability of Data:** ISOs and RTOs regularly publish data regarding the transmission level grid operations online for public use [6][7]. However, no such framework exists for distribution systems. Hence, the only readily available distribution system level datasets are through the efforts of various researchers [8], which are limited in scope.
- **Scale of Data:** Several machine learning based models require vast amounts of data for training. The limited scope and size of the available public datasets prevents the application of sophisticated models to obtain highly accurate results.
- **Privacy of Data:** Distribution system data, obtained from AMI meters contain Personally Identifiable Information (PII) and sophisticated algorithms are required to anonymize the data as per the regulations [9]. This further prevents the large scale availability of such datasets.

An intuitive way to approach these problems is to generate synthetic datasets that enable researchers to develop novel data-driven models, while maintaining real dataset privacy. Classic approaches involve modeling underlying causes of the observed dataset and generating model-based synthetic data. In [10], the authors propose a multisegment markov chain model of the solar states and generate synthetic states using this model. In [11], the author proposes to train autoregressive models and use theta-join for generating smart meter data. However, their approach requires hand-crafted features such as fluctuations flattening and time series deseasonalizing. Accurate modeling of the underlying causes is a daunting task. It requires us to make several assumptions (for example, markovian property) which are not necessarily true, thus affecting the *reliability* of the synthetically generated data.

Potential applications in smart grid that can benefit from large scale synthetic data includes behind-the-meter solar disaggregation [3], real-time smart grid system simulation [12] and etc. In behind-the-meter solar disaggregation problem, large scale datasets are required to train and validate machine learning models, yet many of them are not available due to privacy issues. Real-time smart grid system simulation also requires large scale datasets that reflect certain system behavior, which is often limited due to the lack of fine-grained meters.

In this work, we develop a novel data-driven approach for generating synthetic smart grid data by directly 'learning' the probability distribution of the real time-series data using a deep Generative Adversarial Network (GAN) model. While GANs have been used to effectively synthesize cutting-edge "fake" images and audios [13], they have not hitherto been used for smart grid data due to various underlying challenges in distinguishing data patterns (seasonality, short/long term, customer behavior, prosumer etc.). Our work is based on the following insight: we observe that smart grid time series data can be separated into two distinct statistical components: **Level** and **Pattern**, where Level determines high-level

statistical attributes such as mean, scale and variance while Pattern determines the real trend. By normalizing the Level of different users, the Pattern of long-term periodic time series data can be modeled as a conditional probability distribution conditioned on actual date. We show that this probability distribution can be easily "learned" using GAN. The main contributions of this paper are as follows:

- We first develop a probabilistic model to abstract significant characteristics inherent in smart grid time series datasets.
- We then develop a conditional GAN to learn the probability distribution of the real dataset in order to generate synthetic datasets which are indistinguishable under statistical tests. To the best of our knowledge, this is the first effort that uses deep GANs in smart grid.
- We evaluate the effectiveness of the generated synthetic datasets by performing both statistical tests as well as classic machine learning tasks including timeseries clustering and load prediction and showing that the results are indistinguishable from the real dataset.

II. Background

A. Target of Smart Grid Dataset Definition

The immense heterogeneity in smart grid data makes it highly unlikely that a single model could be used for synthesizing the datasets. Hence, any discussion on synthetic dataset generation is incomplete without a precise definition of the targeted datasets. The datasets that we target in this work can be broadly defined as "timeseries data conditioned on smart grid". More precisely, we focus on the datasets which can be modeled as a timeseries. Moreover, the underlying processes generating the dataset should be defined or affected by the smart grid under consideration. This implies that data generated using natural processes such as temperature, solar irradiance etc. are not our focus. Moreover, event based data [14] such as on/off time of appliances, plug in/plug out times of EVs etc. are also not a focus.

The above definition allows us to model (trivially) a wide range of datasets such as uncontrolled load: affected by customer behavior patterns which are conditional on economic features etc.; PV generation: affected by solar irradiance assuming no forced curtailment [15] and conditional on PV module number, size, efficiency; aggregate generation in the grid: affected by smart grid demand and thus conditional on smart grid in consideration; electricity prices: affected by market conditions and thus conditional on the smart grid in consideration etc. We can also model controlled load/generation (e.g. due to Demand Response [16], load/solar curtailment etc. [15]) by adding additional conditional variables to denote the control. For example, if a building has two different consumption profiles, one under normal conditions and one under DR, then an additional binary conditional variable to denote whether the building is in DR or not can be used.

B. Generative Adversarial Network

GAN [13] is a deep generative model that can implicitly capture any differentiable probability distribution and provide a way to draw samples from it. Assuming some prior distribution $z \sim p_z(z)$, we would like to learn a generator function G, such that $G(z) \sim p_{data}(x)$. To achieve this goal, we introduce a discriminator function D and let D and G play the following two-player minimax game with value function $V(G, D)$:

$$\min_G \max_D V(G,D) = \mathbb{E}_{x\sim p_{data}(x)}[\log D(x)] + \mathbb{E}_{z\sim p_z(z)}[\log(1 - D(G(z)))] \quad (1)$$

An intuitive explanation of the objective function is that the generator is trying to produce fake samples while the discriminator is trying to detect the counterfeits. A competition in this game drives both the generator and the discriminator to improve their methods until the fake samples are indistinguishable from the real data.

In practice, function G and D are often approximated using deep neural networks. It is proven in [13] that given fixed discriminator D, minimizing the value function in Eq 1 with respect to the generator parameters is equivalent to minimizing the Jensen-Shannon divergence between $p_{data}(x)$ and $G(z)$. In other words, as training progresses, the implicit distribution $G(z)$ captures converges to $p_{data}(x)$. In this work, we use a variant GAN architecture known as Conditional GAN [17] to learn conditional probability distribution of time series data.

C. Evaluating Synthetic Datasets

As it is not possible to mathematically prove that the real samples and the synthetic samples come from the identical distribution, we perform statistical tests and use classic machine learning algorithms to empirically show:

1) Real time series and synthetic time series share key statistical properties.
2) Real time series and synthetic time series **can not be distinguished** by the outcomes of these machine learning algorithms.

Ultimately, the purpose of the synthetic data is to serve as supplemental training data for machine learning solutions or as a substitute data to preserve privacy of the original dataset.

1) Statistical Tests: Maximum Mean Discrepency (MMD) [18] measures the distance between two probability distributions by drawing samples. Given samples $\{x_i\}_{i=1}^N \sim p(x)$ and $\{y_j\}_{j=1}^M \sim q(y)$, an estimate of ***MMD*** is:

$$MMD = \left\{ \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} K(x_i, x_j) - \frac{2}{MN}\sum_{i=1}^{N}\sum_{j=1}^{M} K(x_i, y_j) + \frac{1}{M^2}\sum_{i=1}^{M}\sum_{j=1}^{M} K(y_i, y_j) \right\}^{1/2} \quad (2)$$

where $K(x, y) = \exp(-||x - y||^2/(2\sigma^2))$ is known as radial basis function (RBF) kernel.

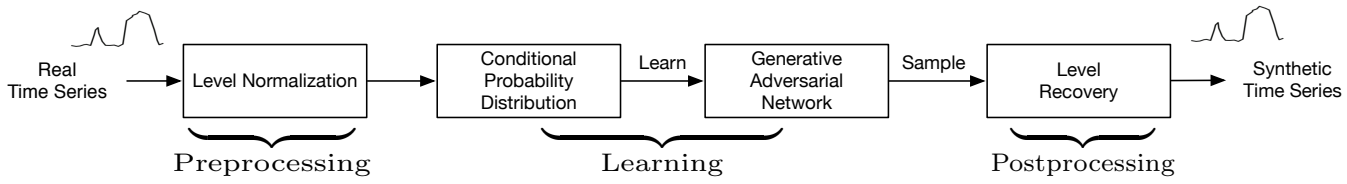


Fig. 1: Data Processing Flow

2) Machine Learning Algorithms: We perform classic machine learning tasks e.g. clustering and forecasting on both original and synthetic dataset. A potentially useful synthetic dataset shall show similar results as the original dataset in these tasks.

III. Approach

A. Probabilistic Modeling for Time Series Data in Smart Grid

In smart grids, human activity datasets often contain daily, monthly and seasonal patterns. They arise from the periodic nature of human behavior. We assume that each time series is the sum of two components: **Level** and **Pattern**.

Level is determined by attributes such as household consumption level, season etc. and affects the *scale* and *bias* in time series data. For example, high income households probably consume more energy than low income households; all households usually consume more energy in July than in October due to high temperature HVAC use. **Pattern** is determined by household activity. Office worker households may consume very little daytime energy as people are out for work; night-owl households consume more energy in night time. In this work, we assume Level is captured by daily mean and standard deviation of the time series and we use GAN to learn user Patterns.

We split the entire time series into daily vectors. Each daily data vector of a particular user u is denoted as $x_{u,t} = (x_{u,t1}, x_{u,t2}, \cdots, x_{u,ts})^T$, where s is the number of samples per day and t is the index of day. We assume that the distribution of daily data vector is conditional dependent on the day index t and the user index u. Furthermore, we assume the day index t can be represented by the day of the week $\{Sun, Mon, \cdots, Sat\}$ and the month of the year $\{Jan, Feb, \cdots, Dec\}$. Mathematically, we can write

$$x_{u,t} = a_{u,t}x^{'}_{u,t} + b_{u,t} \tag{3}$$

where $a_{u,t}$ and $b_{u,t}$ captures the **Level** information ($a_{u,t}$ represents the scale and $b_{u,t}$ represents the bias) and $x^{'}_{u,t}$ captures the **Pattern** information. Then, the conditional probability distribution is denoted as

$$x^{'}_{u,t} \sim p(x^{'}_{u,t}|day\ of\ week, month, user) \tag{4}$$

Each user has different behavior. Thus the time series produced by different users are sampled from different distributions. Given training data from N different users $\{x_{u,t}\}_{t=1}^{T}$, where $u = 1, 2, \cdots, N$, the goal is to train a generator function G that can produce samples subject to distribution p without explicitly modeling or calculating p.

B. Data Processing Flow

We show the data processing flow in Figure 1. In preprocessing step, we perform level normalization to obtain conditional probability distribution as per the following equation:

$$x^{'}_{u,t} = \frac{x_{u,t} - b_{u,t}}{a_{u,t}} \tag{5}$$

where $a_{u,t} = \sigma_{x_{u,t}}$ and $b_{u,t} = \overline{x_{u,t}}$ is the standard deviation and the average of the energy consumption or the solar generation of user u in day t, respectively. In the learning phase, we use GAN to implicitly learn the distribution p and generate synthetic samples $\hat{x}^{'}_{u,t}$. In postprocessing step, we perform level recovery as:

$$\hat{x}_{u,t} = \hat{x}^{'}_{u,t} \times \sigma_{x_{u,t}} + \overline{x_{u,t}} \tag{6}$$

IV. Experimental Setup

A. Dataset Description

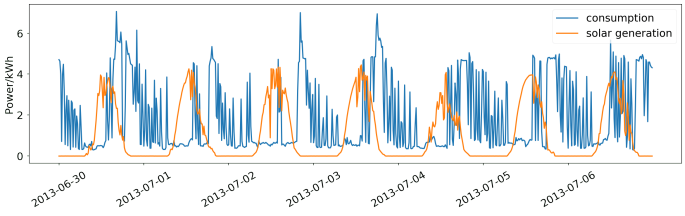
We conduct experiments using **Pecan Street Dataset**, which is free for university researchers [5]. We use a subset which records energy consumption and solar generation of 25 users with PV panels installed. The dataset is collected from 2013-01-01 to 2016-12-31 by averaging consumption and solar generation within each 15 minute. We show the energy consumption and solar generation of user 93 from 2013-10-08 to 2013-10-14 in Figure 2. We observe that solar generation shows repetitive patterns with noise originated from cloud and rain. Consumptions also have similar trends within this week, but with irregular spikes.

B. GAN Architecture

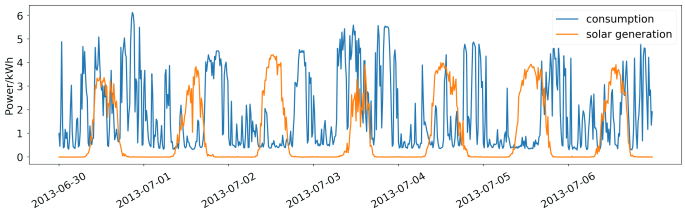
We show our conditional GAN architecture in Fig 3. We use embedding layer to transform one-hot day and month labels into vector representation. The generator concatenates day vector, month vector and noise sampled from normal distribution, feeds them into 3-layer 1D tranpose convolutional network and produces the synthetic output. The discriminator is a 3-layer 1D convolutional network that takes real or synthetic data, day vector and month vector as input, produces a label of whether the input is real or synthetic.

C. Machine Learning Algorithms for Evaluation

For each user $u = 1, 2, \cdots, N$ in the real dataset $\{x_{u,t}\}_{t=1}^{T}$, we generate synthetic data $\{\hat{x}_{u,t}\}_{t=1}^{T}$ of the same time length (4 years with 15 minutes interval in Pecan Street Dataset).



(a) User 93 Real Consumption and Generation



(b) User 93 Synthetic Consumption and Generation

Fig. 2: User 93 Real and Synthetic Data from 2013-06-30 to 2013-07-06.

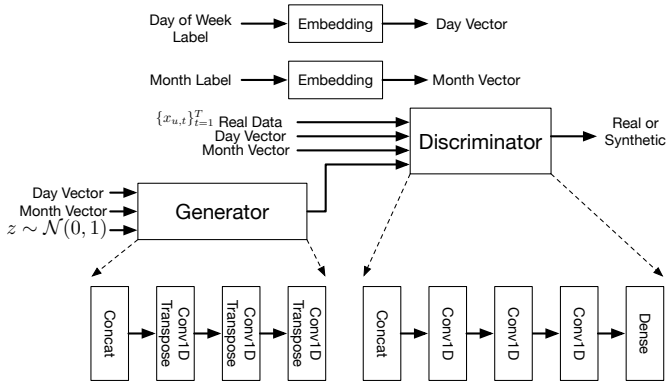


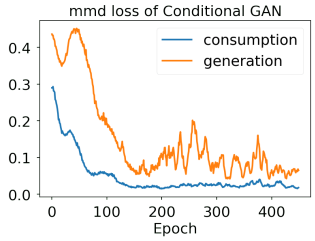
Fig. 3: Conditional GAN architecture

1) K-means Clustering: We perform dynamic time warping (DTW) k-means clustering [19] with data in January 2013 to cluster users according to their consumption and generation patterns. We conduct 3 sets of experiments as follows:

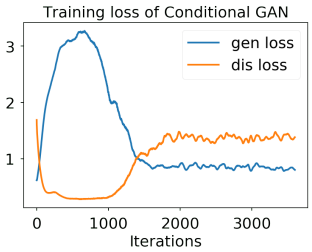
- Fit a model using real data and predict the cluster labels on synthetic data.
- Fit a model using synthetic data and predict the cluster labels on real data.
- Fit a model using mixed real and synthetic data.

Ideally, real time series and synthetic time series generated from the same user should belong to the same cluster. We compare the real labels and synthetic labels using F1-score [20]. In this experiment, we set the number of clusters K to be 3 to simplify the illustration of the results.

2) Short-term Load Forecasting: We perform short-term load forecasting to demonstrate the statistical similarities between real data and synthetic data using **Auto-Regressive Integrated Moving Average (ARIMA)** [21] model. We predict the next 24-hour consumption based on the previous week data of a single user. We conduct two sets of experiments on consumption and solar generation, respectively:



(a) Maximum Mean Discrepancy



(b) Training Loss

Fig. 4: Statistics during Training

- Train on real data and test on real data
- Train on synthetic data and test on synthetic data

We run the prediction 100 times in each experiment on different days and compare the prediction performance using **mean absolute percentage error (MAPE)**.

ARIMA model is defined in terms of three parameters [22]:

- p: the auto-regressive order that denotes the number of past observations included in the model.
- d: the number of times a time series needs to be differenced to make it stationary.
- q: the moving average order that denotes the number of past white noise error terms included in the model.

In training ARIMA model, the parameters were (5, 1, 0).

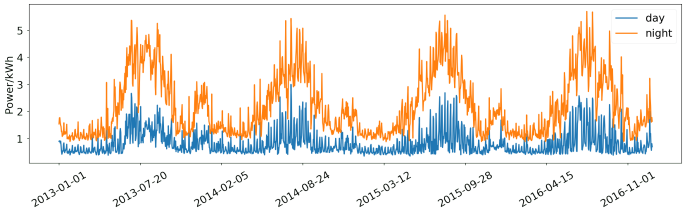
V. Experimental Results

A. Statistical Property Analysis

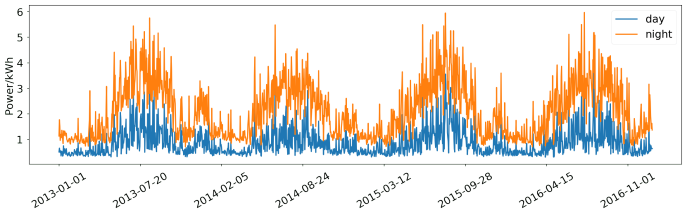
1) Maximum Mean Discrepancy: We use 3 years data from 2013-01-01 to 2015-12-31 to train a Conditional GAN and use data from 2016-01-01 to 2016-12-31 as validation data. As show in Figure 4b, the generator loss and discriminator loss converges, indicating that the GAN approaches Nash Equilibrium after training for 2000 iterations. After each training epoch, we generate 1 year synthetic data and compute *Maximum Mean Discrepancy (MMD)* against the validation data. We show the MMD curve in Figure 4a. We observe that the MMD gradually decreases and converges as training goes on. Thus, the probability distribution of the synthetic data approaches the real data distribution and finally converges.

2) Real Data vs. Synthetic Data: We plot a week of the real data and the synthetic data of user 93 in Figure 2. We summarize key properties of the real data that GAN captures in synthetic data:

- **Data Range:** The range of the real and the synthetic data matches as shown in Figure 2. The peak consumption is around 6 kW and the peak solar generation is around 4 kW. This is critical because it is a necessary condition of the indistinguishability between the real data and the synthetic data.
- **Day Time Consumption vs Night Time Consumption:** We observe that households with PV panels installed typically consume less energy in day time than night time. We plot 4 years average day time consumption and



(a) Real consumption during day time and night time



(b) Synthetic consumption during day time and night time

Fig. 5: Four years average day time and night time consumption of a user. Day time is 6am to 6pm. Night time is 12am to 6am plus 6pm to 12am.

TABLE I: K-means clustering prediction results

Train	Test	F1 Score
Real	Synthetic	1.00
Synthetic	Real	1.00
Mixed	Mixed	0.96

night time consumption of a user in Figure 5. As shown in Figure 5, the synthetic data captures the general pattern as shown in real data. However, the synthetic data is noisy than real data.

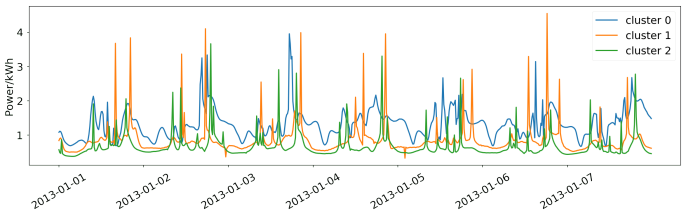
- **Solar Generation Noise:** The solar generation is proportional to the solar radiation when it is sunny. The noise (glitches) on solar generation curve are caused by cloud or rain during sampling period. We notice that the synthetic data automatically captures this feature as shown in Figure 2.

B. Time Series Clustering

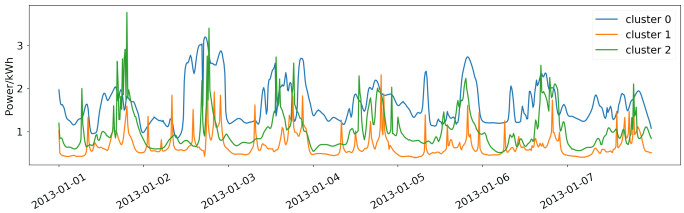
We show the centroids of 3 clusters trained on real data, synthetic data and mixed data in Figure 6.

Insight 1: Real time series and synthetic time series of the same user belong to the same cluster. We report the F1 score of various settings. The high F1 score indicates that the real data and synthetic data from the same user belongs to the same cluster. This matches our hypothesis mentioned in Section III-A.

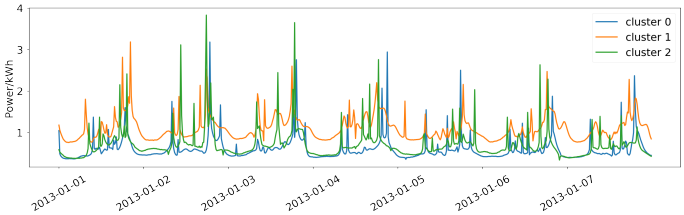
Insight 2: The primary criteria of clustering is the consumption level. In all three centroids subfigures, there is a centroid with apparently higher consumption level than the other two clusters. The other two centroids fit on synthetic data can also be separated by consumption level. However, this is not true for centroids trained on real data and mixed data.



(a) K-means centroids fit on real data

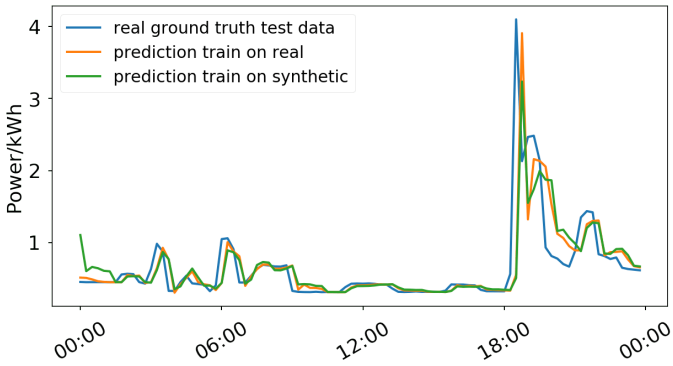


(b) K-means centroids fit on synthetic data

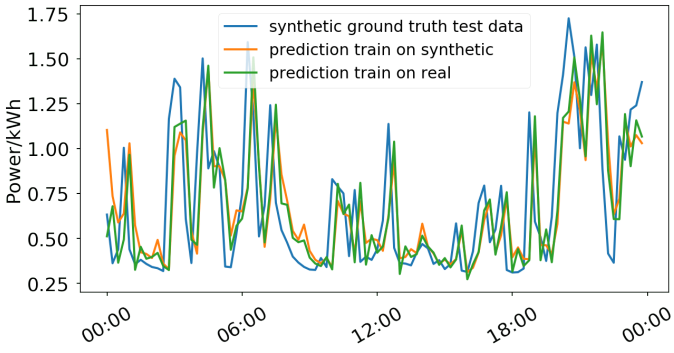


(c) K-means centroids fit on mixed data

Fig. 6: K-means centroids of various settings. We only show one week curve for demonstration.



(a) Prediction on real load



(b) Prediction on synthetic load

Fig. 7: Short-term load forecasting results

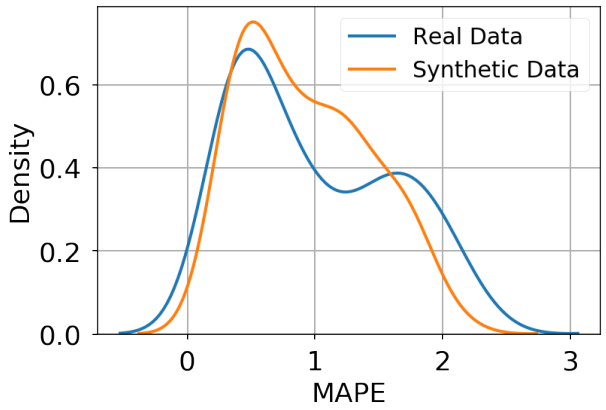


Fig. 8: Mean absolute percentage error (MAPE) distribution of load prediction using real data and synthetic data

C. Short-term Load Forecasting

We show one day of load prediction result in Figure 7. The predicted curve generally matches the ground truth on both real data and synthetic data. We show the mean absolute percentage error (MAPE) density distribution of 100 predictions on real and synthetic data in Figure 8. We observe that MAPE of both curves have concentration near $0.5 \sim 0.6$. It provides empirical grounds for the statistical identity of the real data and the synthetic data. We also notice that the MAPE predicted using real data has more variance error due to higher noise.

VI. Related Work

Data-driven approach for generating synthetic time-series data has been widely studied. In [23], the author proposes to use Hidden Markov Model to generate synthetic time series data. The HMM-based approach makes strong assumption of the Markovian property of the time series data, which may not hold. While in our approach, the assumption of conditional probability distribution is much more relaxing. In [24], the author successfully applied Recurrent Neural Network GAN to synthesize real-value medical signal data. The medical signal contains auto-regressive property while in daily user consumption, the value at a certain timestamp does not depend on the previous timestamp. That's why we use Convolutional Nets to capture the daily vector patterns instead of using RNN to capture the auto-regressive patterns.

VII. Conclusion

Lack of fine grained time series datasets in distribution level smart grid prevent the advance of new data-driven methods. In this paper, we established a general probabilistic time-series data model. We proposed to use generative adversarial network to produce synthetic datasets that sampled from the same distribution as the real datasets. To evaluate the synthetic daatsets, we perform statistical tests as well as classic machine learning tasks including time series clustering and load prediction. Empirical results show that the synthetic datasets and real datasets are not distinguishable.

Acknowledgment

This work has been supported in part by the U.S. National Science Foundation under EAGER Award No.:CNS-1637372 and the U.S. Department of Energy (DoE) under award number DE-EE0008003.

References

[1] A. K. Srivastava *et al.*, "Short-term load forecasting methods: A review," in *2016 International Conference on Emerging Trends in Electrical Electronics Sustainable Energy Systems (ICETEESES)*, March 2016, pp. 130–138.

[2] S. Aman *et al.*, "Prediction models for dynamic demand response: Requirements, challenges, and insights," in *2015 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, Nov 2015, pp. 338–343.

[3] D. Chen and D. Irwin, "Sundance: Black-box behind-the-meter solar disaggregation," in *Proceedings of the Eighth International Conference on Future Energy Systems*, ser. e-Energy '17. New York, NY, USA: ACM, 2017, pp. 45–55.

[4] S. Barker *et al.*, "Smart*: An open data set and tools for enabling research in sustainable homes."

[5] "Pecan street dataset," http://www.pecanstreet.org/category/dataport/, 2018.

[6] "Pjm dataset," http://www.pjm.com/markets-and-operations/.

[7] "New york system independent system operator," https://www.nyiso.com/public/markets_operations/market_data/.

[8] Y. Wang *et al.*, "Review of smart meter data analytics: Applications, methodologies, and challenges," *CoRR*, vol. abs/1802.04117, 2018.

[9] M. R. Asghar *et al.*, "Smart meter data privacy: A survey," *IEEE Communications Surveys Tutorials*, vol. 19, no. 4, pp. 2820–2835, Fourthquarter 2017.

[10] W. Tushar *et al.*, "Synthetic generation of solar states for smart grid: A multiple segment markov chain approach," in *IEEE PES Innovative Smart Grid Technologies, Europe*, Oct 2014, pp. 1–6.

[11] N. Iftikhar *et al.*, "A scalable smart meter data generator using spark," in *On the Move to Meaningful Internet Systems. OTM 2017 Conferences*, H. Panetto *et al.*, Eds. Cham: Springer International Publishing, 2017, pp. 21–36.

[12] F. Guo *et al.*, "Comprehensive real-time simulation of the smart grid," *IEEE Transactions on Industry Applications*, vol. 49, no. 2, pp. 899–908, March 2013.

[13] I. Goodfellow *et al.*, "Generative adversarial nets," in *Advances in Neural Information Processing Systems 27*, Z. Ghahramani *et al.*, Eds. Curran Associates, Inc., 2014, pp. 2672–2680.

[14] K. Anderson *et al.*, "Blued: A fully labeled public dataset for event-based non-intrusive load monitoring research," pp. 1–5, 01 2012.

[15] S. R. Kuppannagari *et al.*, "Optimal net-load balancing in smart grids with high PV penetration," *CoRR*, vol. abs/1709.00644, 2017.

[16] R. Deng *et al.*, "A survey on demand response in smart grids: Mathematical models and approaches," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 3, pp. 570–582, June 2015.

[17] M. Mirza and S. Osindero, "Conditional Generative Adversarial Nets," *ArXiv e-prints*, Nov. 2014.

[18] D. J. Sutherland *et al.*, "Generative models and model criticism via optimized maximum mean discrepancy," *ICLR*, 2017.

[19] S. Salvador and P. Chan, "Toward accurate dynamic time warping in linear time and space," *Intell. Data Anal.*, vol. 11, no. 5, pp. 561–580, Oct. 2007.

[20] M. Sokolova *et al.*, "Beyond accuracy, f-score and roc: A family of discriminant measures for performance evaluation," in *AI 2006: Advances in Artificial Intelligence*, A. Sattar and B.-h. Kang, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 1015–1021.

[21] M. T. Hagan and S. M. Behr, "The time series approach to short term load forecasting," *IEEE Transactions on Power Systems*, vol. 2, no. 3, pp. 785–791, Aug 1987.

[22] G. E. P. Box and G. Jenkins, *Time Series Analysis, Forecasting and Control*. Holden-Day, Incorporated, 1990.

[23] M. Arlitt *et al.*, "Iotabench: an internet of things analytics benchmark," pp. 133–144, 01 2015.

[24] C. Esteban *et al.*, "Real-valued (medical) time series generation with recurrent conditional gans," 2017.