RESEARCH PAPER

Generative adversarial network for load data generation: Türkiye energy market case

Bilgi Yilmaz [iD] [1,2,*,‡]

[1]Department of Mathematics, Kaiserslautern Technical University, Kaiserslautern, Germany,
[2]Department of Mathematics, Faculty of Science, Kahramanmaras Sutcu Imam University, Türkiye

*Corresponding Author
‡yilmazb@rhrk.uni-kl.de (Bilgi Yilmaz)

Abstract

Load modeling is crucial in improving energy efficiency and saving energy sources. In the last decade, machine learning has become favored and has demonstrated exceptional performance in load modeling. However, their implementation heavily relies on the quality and quantity of available data. Gathering sufficient high-quality data is time-consuming and extremely expensive. Therefore, generative adversarial networks (GANs) have shown their prospect of generating synthetic data, which can solve the data shortage problem. This study proposes GAN-based models (RCGAN, TimeGAN, CWGAN, and RCWGAN) to generate synthetic load data. It focuses on Türkiye's electricity load and generates realistic synthetic load data. The educated synthetic load data can reduce prediction errors in load when combined with recorded data and enhance risk management calculations.

Keywords: Load in Türkiye energy market; generative adversarial networks; synthetic data generation; unsupervised learning; RCGAN; TimeGAN; CWGAN; RCWGAN

AMS 2020 Classification: 68T07; 82C32; 68T05

1 Introduction

A *smart grid* is an electricity distribution network incorporating information and communication technologies to improve energy efficiency. It allows for the real-time exchange of data between electricity suppliers and consumers, which enables suppliers to forecast electricity demand based on current energy consumption and user profiles. This feature enables energy suppliers to optimize electricity efficiency by providing accurate load modeling, resulting in a more efficient power grid [1].

In the context of smart grids, accurate electricity demand forecasting is crucial for energy suppliers to avoid financial losses and system troubles, e.g., drops in frequency and blackouts. However,

obtaining an extensive and high-quality electricity dataset is challenging and expensive. Although electricity grid models are known, data is short as a consequence of privacy concerns, which restricts the researchers' access to datasets and limits the development and application of further load prediction models.

Generating synthetic data that accurately represents real data's statistical behavior and characteristics can help address subjects connected to sensitive data's quantity, quality, and privacy. By generating synthetic data that mimics the patterns and trends of real data, it is possible to provide researchers and companies with a valuable resource for understanding the distribution of the original data while also enabling efficient data storage, data augmentation, system testing, and data disclosure. Synthetic data generation can also help mitigate concerns around data privacy, as it can provide a substitute for sensitive or confidential data that is not accessible to third parties. This study focuses explicitly on Generative Adversarial Networks (GANs) for generating synthetic data regarding their implementation performance and flexibility in mirroring historical data. GANs have successfully generated and manipulated images and natural languages, as demonstrated by various studies [2, 3, 4, 5]. As a result, GANs have become a prominent method for synthetic data generation.

GANs are powerful generative models that can assemble new samples having similar distributional properties to the real data, making them useful for data augmentation [6]. While initially developed for image processing and computer vision, GANs have garnered significant interest and advanced in various research fields [7]. GANs also demonstrated favorable outcomes in generating sequential data (e.g., music, medical data, and finance). Therefore, this study focuses on applying GANs to sequential data, specifically generating synthetic load data for Türkiye energy market.

There are two primary strategies for applying GANs to electricity consumption data forecasting. The first strategy uses a typical GAN architecture in generating synthetic load data. The performance is evaluated concerning the divergence or convergence of synthetic data to real data. The second strategy involves using more complex GAN architectures to generate synthetic electricity consumption data and combining it with real data to expand and improve real load data. The first approach is limited to scenario generation. It produces load profiles lacking precision, while the second approach is data augmentation, which is highly influential but must fully illustrate the capabilities of GANs. Therefore, this study focuses on the first strategy of synthetic data generation using GANs to produce hourly electricity consumption records.

Although usually, studies suggest using Long Short-Term Memory (LSTM) in GANs, this study avoids utilizing it to decrease the computational cost of training and overfitting problems. Instead, the study uses Recurrent Neural Network (RNN). More specifically, it uses the GANs called Recurrent Conditional GAN (RCGAN) [8], Time-Series GAN (TimeGAN) [9], Conditional Wasserstein GAN (CWGAN) [2], and Recurrent Conditional Wasserstein GAN (RCWGAN) as in [10].

The remaining part of the study is systematized as follows: Section 2 briefly reviews the literature on GANs. Section 3 presents an overview of the GANs used in this study without delving into technical facts. Section 4 introduces Türkiye's load data and includes exploratory data analysis and synthetic data generation using the selected GANs. Section 5 concludes the study.

2 Literature review

The concept of GANs was submitted in the paper by [11] and quickly gained traction in many research fields, such as 3D object generation [12], electronic health record generation [8], image processing [13] and generation [11, 14], face detection [15], audio synthesis [16, 17] natural language processing [18], traffic controlling [19], energy market modeling [10], and stock market modeling [20]. However, training GANs are challenging since they generally suffer from missing

modes problem or model collapse, where the generated samples lack variety and only cover some regions of the space. Another common issue is vanishing gradients that can stop GANs training from converging to an optimal state. Dealing with these problems has gained significant attention in GAN research, and various approaches have been proposed to mitigate them.

Consequently, many alternative GANs are developed from empirical and mathematical perspectives to solve such problems. For instance, the study of [2] is the first study that extends the GANs by comparing various distance measures and suggests using the Wasserstein-1-metric, which leads to the development of WGAN. Later, [21] proposed the Least Squares GANs (LSGANs) that adopt a least-squares loss function for the discriminator to overcome the vanishing gradients problem. [22] also proposed a conditional model that extends the original GAN framework by incorporating additional information such as labels, tags, or attributes. This information is provided to the GAN framework through an additional input layer. However, it is outside the content of this study to provide a complete review of such approaches. Therefore, the study covers only the studies focusing on time series data since it is interested in generating time series data, and it can draw some inspiration from applying GANs in financial and electricity markets.

Various GANs have been proposed to generate financial and energy time series data. Financial time series are more challenging to model than other time series because of their high volatility and unexpected market behavior. Therefore, alternative GAN models have been proposed to overcome these challenges. For instance, one of the earliest works presented by [20] offers QuantGANs for financial time series data generation. The QuantGANs utilizes temporal convolutional networks to capture long-range dependencies and can generate realistic stock price simulations employing a data-driven neural network. The QuantGANs can capture the temporal dependence of financial time series, including volatility clustering. [23] proposed a variant of Conditional GANs (cGANs), called Stock-GAN, to generate order flow in the limit order book. The authors showed that cGANs could generate a realistic and high-fidelity stock market. Similarly, [24] and [25] generated transaction prices in a stock market by using cGAN and illustrated the accuracy of GANs in stock markets.

Recently, the use of GANs in electricity markets has gained significant attention. [10] utilized RCGAN, TimeGAN, CWGAN, and RCWGAN for univariate electricity consumption time series data generation. In their empirical analysis, the authors showed that all four GANs could generate realistic electricity consumption for an individual. Furthermore, they showed the GANs' stability and no vanishing gradient. [26] employed Deep Convolutional GANs (DCGANs) to generate power profile scenarios for wind and solar power plants and energy consumption data. They show that GANs captured the patterns of renewable energy production in both temporal and spatial dimensions under the assumption of a large number of correlated resources. [27] utilized cWGAN to generate synthetic energy consumption data and generated realistic energy consumption data by given labels as a condition imitating real data distribution. [28] used deep learning GANs in generating electricity consumption and fault diagnosis to develop smart management tools for heating, ventilation, and air conditioning (HVAC). The authors showed that deep learning GAN can help to increase the fault diagnosis accuracy in electricity consumption.

[29] proposed GANs as a novel method to generate realistic electrical load profiles of buildings. They showed that the load profiles generated by GANs could mirror the general load trend and the random variations of the actual loads in buildings. Furthermore, they suggested that GANs detect changes in load profiles, anonymize smart meter data, and support grid management applications. [30] utilized GANs to quantify the uncertainties related to the climate and human-system-driven in the energy market. They revealed that climate-driven uncertainties in human systems cause higher fluctuations in load profiles. [31] also aimed to build scenarios by embedding GANs and understanding the stochastic and dynamic characteristics of renewable energy re-

sources. They demonstrated that GANs achieved the controllable generation of renewable energy generation scenarios covering various statistical characteristics and revealed new patterns. [32] benefited GANs to generate scalable and realistic energy demand. The authors claimed that GANs are promising for generating realistic energy demand data. [33] offer GANs a potential approach for predicting large-scale building energy consumption to manage grid operations. To this end, [33] used various GANs (the original GAN, cGAN, SGAN, InfoGAN, and ACGAN) to predict large-scale building energy consumption. The authors claimed that the success of the GANs highly depends on the data size. Further, they claim that SGAN and InfoGAN are unsuitable for large-scale building electricity consumption prediction since these two GANs do not control the number of generated building samples for different building types.

Machine learning applications in Türkiye's energy market are not a new concept, and there are remarkable works in the literature. For instance, [34] used artificial neural networks (ANN) to predict and forecast energy consumption and make correct investments in Türkiye by considering economic indicators (gross national product—GNP and gross domestic product—GDP) and population increase as independent variables. [35] used ANN to forecast electricity consumption in various sectors. Similarly, [36] model Türkiye's energy consumption using ANN and regression analyses to forecast projections by considering explanatory variables, such as socio-economic and demographic factors (gross domestic product (GDP), import and export, population, and employment). [37] and [38] developed acceptable methods based on the ANN model that uses GDP, population, imports, exports, building area, and number of vehicles for estimating Türkiye's future energy demand while [39] developed forecasting models relying on ANN to predict the energy consumption in Türkiye's transportation sector. However, there is no study utilizing GANs for Türkiye's energy market.

3 Generative adversarial networks

Training GANs are challenging since they generally suffer from missing modes problem or model collapse, where the generated samples lack variety and only cover some regions of the space. Another common issue is vanishing gradients that can stop GANs training from converging to an optimal state. Dealing with these problems has gained significant attention in GAN research, and various approaches have been proposed to mitigate them. However, it is outside the content of this study to provide a complete review of such approaches.

As [11] introduced, GANs belong to the family of unsupervised learning algorithms. They can learn dense representations of input datasets and are utilized as generative models. The superiority of GANs is the ability to generate new samples having (nearly) the same distribution as the training dataset. They contain two competing neural networks, Generator (G) and Discriminator (D). Therefore, the training of GANs relies on a zero-sum game. G directly produces samples from a well-known distribution (e.g., normal and uniform distributions) as input (latent vector z), and D attempts to distinguish between samples drawn from training and generated data. The discriminator output ($D(x)$) corresponds to the probability that a sample belongs to the distribution underlying the training data. On the other hand, the generator output ($G(z)$) is a sample from the learned distribution. The competition between G and D is formulated as

$$\min_{G} \max_{D} V(D, G) = \min_{G} \max_{D} \left(\mathbb{E}_{x \sim \mu} \left[\log \left(D(x) \right) \right] + \mathbb{E}_{z \sim \gamma} \left[\log \left(1 - D(G(z)) \right) \right] \right),$$

where $D(x) : \mathbb{R}^n \mapsto [0, 1]$ and $G(z) : \mathbb{R}^d \mapsto \mathbb{R}^n$, where G is the generator function that takes random samples $z \in \mathbb{R}^d$ from a predefined distribution γ (usually a Gaussian distribution) and generates samples $G(z)$ [40, 41]. This linear function illustrates the adversarial competition

between the generator and discriminator. Here, the discriminator outputs a binary variable, where $D(x) = 1$ for real samples and $D(x) = 0$ for generated samples, while the generator outputs a synthetic sample vector.

It is important to note that when it comes to generating or predicting time series data, it is more significant to determine the correct conditional distribution rather than learning the joint distribution. This is because, in predictive modeling, we are concerned with identifying the conditional distributions of the future time series $x_{future} = x_{t+1:t+q}$, which refers to the following q values given the past p observations of the time series $x_{past} = x_{t-p+1:t}$ at time t (for more information, see [42].)

The GANs considered in this study are characterized by the selection of their respective loss functions for the discriminator and generator.

Recurrent Conditional GAN (RCGAN)

The RCGAN shares a similar architecture with the traditional GAN but with a modification where both the generator and discriminator are replaced with recurrent neural networks (RNNs). This change enables the RCGAN to generate sequence data dependent on specific conditional inputs and can produce realistic outputs.

Let $RNN(X)$ be the vector consisting of T outputs from an RNN that receives a sequence of T vectors $\{x_t\}_{t=1}^T$ ($\mathbf{x}_t \in \mathbb{R}^d$), and let $CE(\mathbf{a}, \mathbf{b})$ denote the average cross-entropy between the sequences $\mathbf{a}$ and $\mathbf{b}$. Then, according to [8], the discriminator and generator loss of $\mathbf{X}_n, \mathbf{y}_n$, where $\mathbf{X}_n \in \mathbb{R}^{T \times d}$ and $\mathbf{y}_n \in 0, 1^T$, can be expressed as

$$
\begin{aligned}
D_{loss}(\mathbf{X}_n, \mathbf{y}_n) &= -CE(RNN_D(\mathbf{X}_n), \mathbf{y}_n), \\
G_{loss}(Z_n) &= D_{loss}(RNN_G(Z_n), \mathbf{1}) = -CE(RNN_D(RNN_G(Z_n)), \mathbf{1}),
\end{aligned}
$$

where $\mathbf{y}_n$ is a vector consisting of ones if the sequence is real and zeros if it is fake. Z_n is a sequence of T points drawn from the latent space $\mathbf{Z}$, which is typically a m-dimensional Gaussian distribution. Therefore, Z_n is a matrix with dimensions $T \times m$. The vector $\mathbf{1}$ represents the decision of the discriminator accepting a given sequence as real data. During each training step, the discriminator uses both real and fake sequences.

Time-series GAN (TimeGAN)

TimeGAN was initially introduced in the work by [9]. This approach focuses on datasets that contain both static and temporal features. Static features remain constant and unchanging over time (such as gender), while temporal features change and are updated over time.

The static and temporal features can be represented using the vectors $\mathcal{S}$ and $\mathcal{X}$, respectively. We can also assign specific values to random vectors $\mathbf{S} \in \mathcal{S}$ and $\mathbf{X} \in \mathcal{X}$, represented by s and x, respectively. Let us consider tuples $\mathbf{S}, \mathbf{X}_{1:T}$, where the joint distribution is denoted as p, and the length T of each sequence is also a random variable. In the training data, we can index individual samples using $n \in 1, \dots, N$, and denote the training dataset as $\mathcal{D} = (s_n, x\{n, 1 : T_n\})_{n=1}^N$.

The objective is to find the density $\hat{p}(\mathbf{S}, X_{1:T})$ that satisfactorily approximates the real data density $p(\mathbf{S}, X_{1:T})$ using the training dataset $\mathcal{D}$. However, achieving this task may require more work in the traditional GAN framework. To address this issue, [9] suggests using an autoregressive decomposition of $p((S), \mathbf{X}_{1:T}) = p(\mathbf{S})\Pi p(\mathbf{X}_t \mid \mathbf{S}, \mathbf{X}_{1:t-1})$ to concentrate on the additional information given as conditionals.

TimeGAN is distinct from traditional GANs in that it comprises four neural network components: two autoencoding components, namely the embedding and recovery functions, and two adversar-

ial components, namely the generator and discriminator. The main concept behind TimeGAN is that the autoencoding and adversarial components are trained jointly. Consequently, TimeGAN can simultaneously learn how to encode features, generate replicas, and iterate across time. The embedding network creates the latent space, while the adversarial network operates within this space. By means of a supervised loss, the latent dynamics of both empirical and generated data are synchronized.

Conditional Wasserstein GAN (CWGAN) and Recurrent Condition Wasserstein GAN (RCW-GAN)

The WGAN was first presented in [2] as a solution to address the issues of mode collapse and vanishing gradient in traditional GANs. Instead of optimizing the traditional GAN loss, which is known to be prone to these issues, the WGAN optimizes the Wasserstein-1 distance. However, calculating the exact Wasserstein-1 distance is often impractical, so instead, the objective function is altered to approximate the Wasserstein-1 distance as

$$\min_G \max_D \mathbb{E}_{x \sim p_{data}}\Big[D(x)\Big] - \mathbb{E}_{z \sim p_z}\Big[D(G(z))\Big].$$

If D satisfies a Lipschitz constraint with a constant k, then it can be shown that the Wasserstein-1 distance is equivalent to the supremum of the output difference of D on pairs of inputs. The WGAN uses weight clipping to enforce the Lipschitz constraint, which restricts the weights of D to a compact interval such as $[-c, c]$ where c is a small positive value (e.g., 0.01). However, this technique can limit the capacity of the discriminator. It may cause the weights to converge to the endpoints of the interval, leading to gradient issues like vanishing or exploding gradients.

The WGAN-GP method is an improvement over the WGAN, and it addresses the drawbacks of weight-clipping by using a gradient penalty technique. Weight-clipping is replaced with soft enforcement of the Lipschitz constraint through a penalty on the discriminator. This penalty is based on a differentiable function being 1-Lipschitz if its gradients have a norm of at most 1 everywhere. Therefore, the new objective of the GAN is expressed as

$$\min_G \max_D \mathbb{E}_{x \sim p_{data}}\Big[D(x)\Big] - \mathbb{E}_{z \sim p_z}\Big[D(G(z))\Big] - \lambda \mathbb{E}_{\hat{x} \sim p_{\hat{x}}}\Big[\|\nabla_{(\hat{x})} D(\hat{x})\|_2 - 1)^2\Big],$$

where λ is the penalty coefficient.

The CWGAN is an extension of WGAN-GP that incorporates extra information into the model. This leads to a modified optimization problem that is given by

$$\min_G \max_D \mathbb{E}_{x \sim p_{data}}\Big[D(x \mid y)\Big] - \mathbb{E}_{z \sim p_z}\Big[D(G(z \mid y))\Big] - \lambda \mathbb{E}_{\hat{x} \sim p_{\hat{x}}}\Big[(\|\nabla_{\hat{x}} D(\hat{x} \mid y)\|_2 - 1)^2\Big],$$

where y is the vector of additional information.

The RCWGAN architecture is similar to that of the CWGAN, but instead of using conventional neural networks as the generator and discriminator, Recurrent Neural Networks (RNNs) are employed.

4   Empirical analysis

Data and its stylized facts

The study uses the load data from Türkiye gathered from Epias[1] in its empirical analysis. The data consists of seven years of hourly load in Türkiye's energy market. The hourly load profile data over the period 01.01.2016-31.12.2022 is visualized in Figure 1. The figure shows that the load data contains inherent patterns that can be effectively leveraged through machine-learning techniques for modeling purposes. The load has a strong seasonality and a slightly increasing trend. The figure reveals a significant decrease in electricity consumption in Türkiye in the first and second quarters compared to other years. The decline in electricity demand can be attributed to several factors, such as the interruption of production in plants, reduced work hours resulting from restrictions, and the implementation of lockdown measures in cities. While climatic conditions and industrial activities commonly influence fluctuations in total electricity demand, these regular variations cannot account for the significant decrease observed. [43] explains the sharp decline in electricity demand as a direct consequence of the crisis caused by COVID-19 pandemic, clearly highlighting the impact of the prevailing pandemic conditions. Table 1 summarizes the descriptive
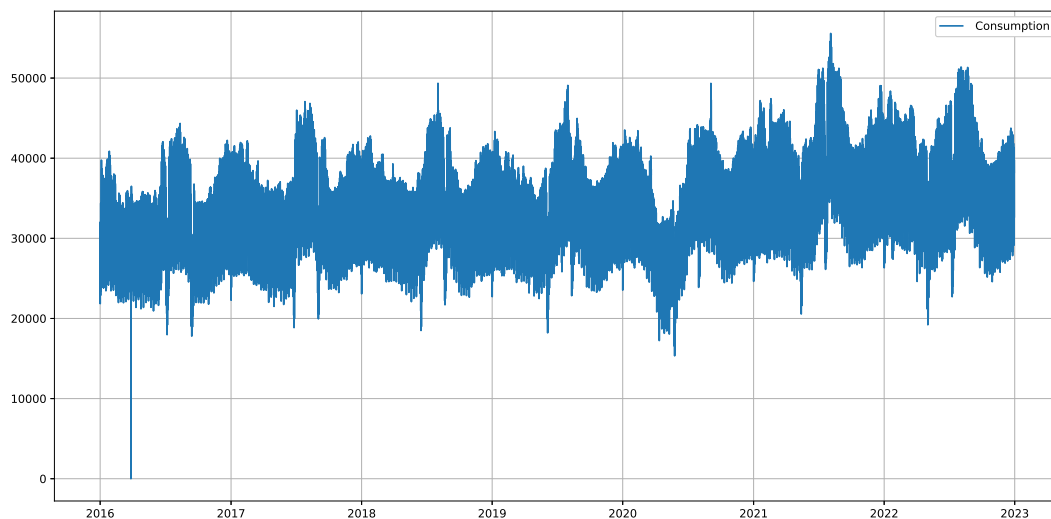
Figure 1. Visualization of load in Türkiye energy market (2016-01-01-2022-12-31)

statistics of hourly load in Türkiye for the considered period. The number of observations is 61345. The average load and Standard Deviation (SD) of load are approximately 34053.14 MWh and 5581.85 MWh in Türkiye's energy market. The SD indicates significant variability in load levels across the considered period. The minimum load is 0 units, which could be an error in the data. The 25% quantile of the observations has a load level below 29875.22 MWh representing the first quartile of the load distribution. 50% of the observations have a load level below 33878.73 MWh. It represents the median of the load data distribution. 75% of the observations have a load level below 38059.42 MWh, which represents the third quartile of the load data distribution.

---

1 https://www.epias.com.tr/

The maximum load is 55575.02 MWh could also be an outlier or a peak value in the investigated period.

**Table 1.** Descriptive statistics of load in Türkiye energy market

	Consumption
count	61345.000000
mean	34053.142402
std	5581.846170
min	0.000000
25%	29875.220000
50%	33878.730000
75%	38059.420000
max	55575.020000

Figure 2 presents the average daily load profile over seven years, 01.01.2016-31.12.2022. The figure shows the variation in electricity consumption over daily, weekly, monthly, and yearly. The figures are arranged sequentially starting from 2016 such that each row represents a specific year; the top figure corresponds to daily electricity consumption in 2016, the second figure from the top illustrates daily electricity consumption in 2017, and finally, the bottom figure presents daily electricity consumption in 2022. The seven rows in each year are days of the week, Monday through Sunday, and lines separate months.

In Figure 2, each cell corresponds to a calendar year day. The color intensity of the cells in the heatmap represents the varying electricity consumption levels across days. The lighter shades indicate higher electricity consumption, while darker shades represent lower electricity consumption. By analyzing this heatmap, we can observe patterns and trends in electricity usage throughout the years. The heatmap helps us to identify peak periods of electricity consumption, such as during weekdays when industrial and commercial activities are at their highest and lower periods during weekends or holidays when there is reduced demand. Such a visualization assists in understanding energy consumption patterns, identifying potential areas for energy conservation, and optimizing electricity distribution and resource planning strategies. For instance, we can detect certain days of the week or times of the year when the load tends to be higher or lower than average. Figure 2 also reveals a significant change in the load patterns over time, such as an increase in demand in August due to changes in the climate. It also reveals a significant decrease in the electricity demand during the weekends and public holidays.

Figure 3 contains distributions of yearly, daily, and monthly electricity consumption data. It provides a visual representation of the distribution of electricity consumption across different periods. The first row illustrates the yearly electricity consumption distributions. It shows the distribution of electricity consumption across each year independently. The box in the middle of the plot represents the interquartile range (IQR) of electricity consumption data, with the median value marked as a line in the box. The whiskers extending from the box represent the range of the data, excluding any outliers. The second row illustrates the distributional properties of electricity consumption in days of the week. It shows the distribution of electricity consumption across each weekday, with the x-axis representing the day of the week and the y-axis representing the electricity consumption level. It reveals any significant change in the electricity consumption on the business days and a slight decrease during the weekends. The third row shows the distributions of electricity consumption for months of the year. It shows that there is an increase in electricity consumption during summers, as in Figure 2, which is most probably due to the cooling needs in the summer. All three figures show that there are outliers in the data indicating
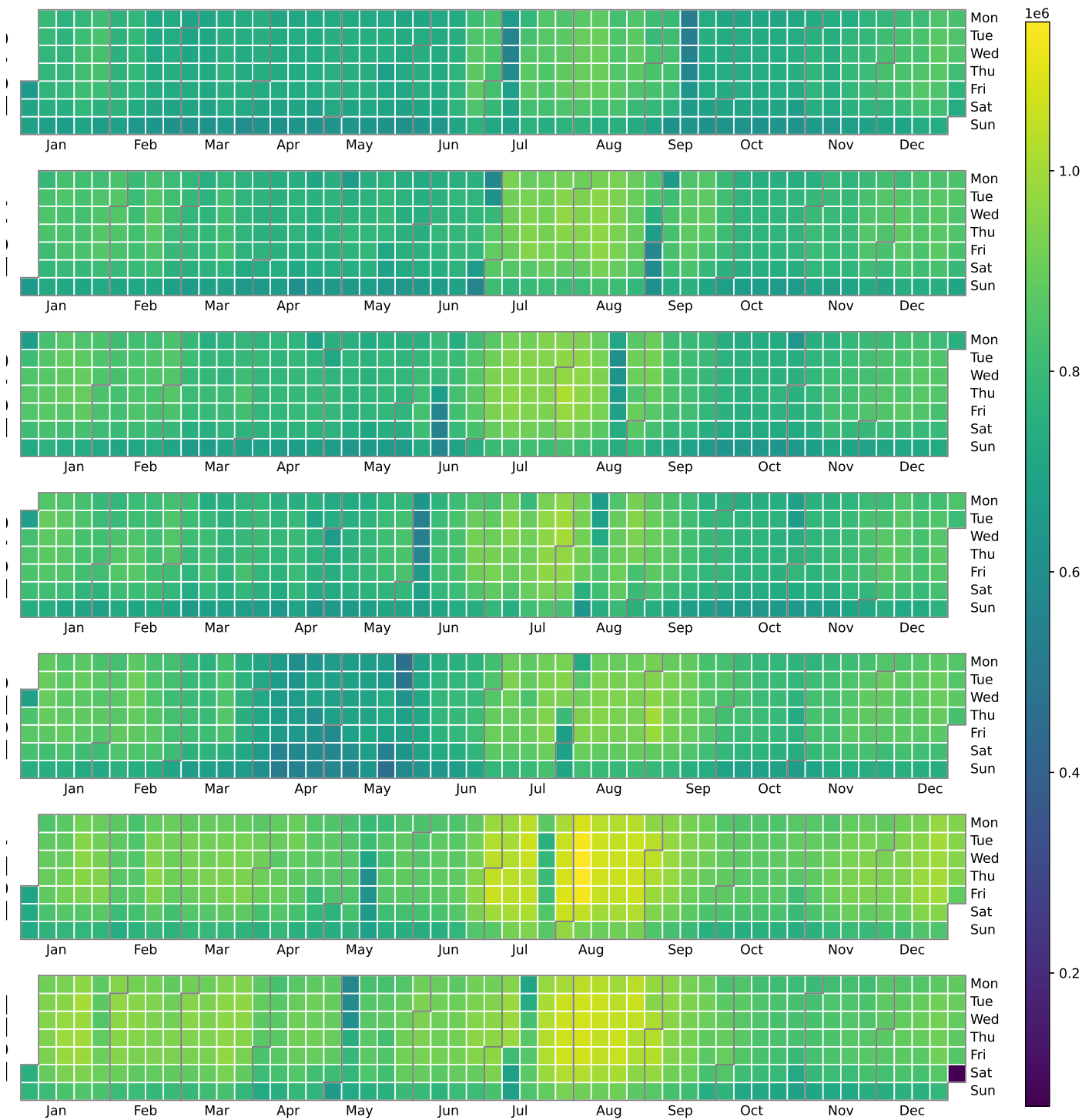
Figure 2. Daily electricity consumption behavior in Türkiye energy market (2016.01.01-2022.12.31)

unusual electricity consumption levels that should be investigated further. In contrast, patterns in the data could reveal important insights into energy usage.

Data preprocessing

In standard modeling and machine learning applications, deseasonalizing the data may be necessary to remove the effects of seasonality. However, deseasonalizing can make it difficult for the GAN to learn the underlying patterns and generate realistic samples since seasonality is an
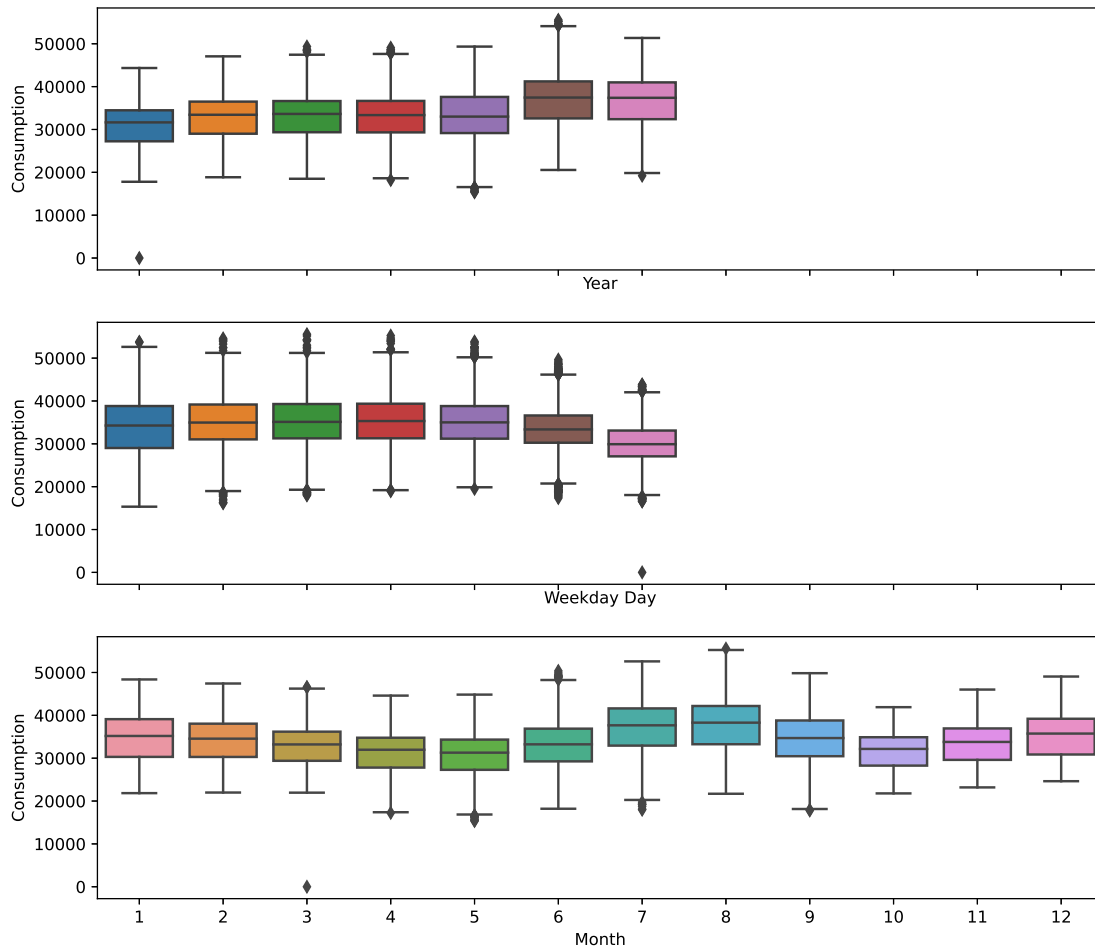
Figure 3. Statistical distribution of load in Türkiye energy market (2016.01.01-2022.12.31)

essential feature of the data we want to preserve. Hence, the data is preprocessed using robust scaling. This scaling method uses the interquartile range. Therefore, it is a robust scaling method for outliers. It has a formula given as

$$y_i = \frac{x_i - Q_{median}(x)}{Q_3(x) - Q_1(x)},$$

where Q_1 and Q_3 correspond to the $1st$ quartile and $3rd$ quartile, i.e., in between $25th$ quantile and $75th$ quantile range, respectively. Consequently, it removes the median and scales the data between Q_1 and Q_3.

Table 2 provides descriptive load statistics after the preprocessing. The length of the training data decreased to 8017. As it is clear, the number of data points is decreased significantly. Here, the study uses only load data in 2017 for the training since the load is quite regular and identical in a classical calendar year (see Figures 2 and 3). After scaling the data, the mean value of -0.045 suggests that the variable has a slightly negative skew, although the value is close to zero. The

standard deviation of 0.677 indicates that the variable has moderate variability. The minimum value of -1.983 and maximum value of 1.864 show the range of load values, with values falling between these two extremes. The quartile values indicate the distribution across the dataset, with the median (50th percentile) value of 0.000 falling at the center of the distribution.

**Table 2.** The descriptive statistics after preprocessing load

	Preprocesed load
count	8017.000
mean	-0.045
std	0.677
min	-1.983
25%	-0.595
50%	0.000
75%	0.405
max	1.864

Experimental studies

The study fixed time series parameters p and q as 4 for the CWGAN and 3 for the remaining GANs to learn the conditional distribution. The discriminators utilize the conditioning time series $x_{t-p+1:t}$ as inputs to generate the part of time series $x_{t+1:t+q}$, i.e., it uses a rolling window size $p + q = 8$ for the CWGAN and $p + q = 6$ for the others. It optimizes GANs algorithms for a total of 1000 generator weight updates. It utilizes the Adam optimizer [44] with parameters $\beta_1 = 0$ and $\beta_2 = 0.9$ to optimize neural network weights in the generator and discriminator and sets the learning rates to 0.001. In the RCGAN and TimeGAN cases, it applies two time-scale updates (TTUR) [45] and sets the learning rate to 0.003. Further, it updates discriminator weights twice per generator weight update to improve stability. The number of epochs operated is 1.000, with a batch size of 200 for all GANs. In the empirical performances, the study uses the Pytorch library [46] to build the GANs. It supplies high-level building blocks for designing deep learning models. Pytorch is a symbolic tensor manipulation framework alternative to TensorFlow.

Figure 4 shows the empirical distributional properties of the GANs and the real data to compare the distributions. The figure reveals a close match between real and synthetic load data distributions. All GANs have relatively close means, skewness, and kurtosis values. The histograms of real and synthetic datasets and their skewness and kurtosis statistics are presented in the figure to measure symmetry, tail behaviors, and changes in their auto-correlation. The histograms in the first column illustrate that the distribution of hourly load from synthetic load data (orange) is nearly equivalent to the real load data (blue). Only the RCGAN has positive skewness statistics, while the remaining GANs have negative skewness statistics, which is also positive for the real data.

In contrast, the kurtosis statistics are all positive for the GANs and real data. The RCGAN has the closes kurtosis statistics (-0.52) to the real data kurtosis statistics (-0.53) while the TimeGAN kurtosis statistics (-0.16) has a more considerable distance. Consequently, the TimeGAN is more peak than the real data. Furthermore, histograms of the log values presented in Figure 4 reveal some discrepancies in the tails of TimeGAN. The CWGAN and RCWGAN are better than the other two GANs in generating low loads, while the TimeGAN is the worst at generating low and high loads. The RCGAN, on the other hand, is the best at generating high load. The auto-correlations of all the GANs and real data are relatively close. While there are some differences in the skewness and kurtosis statistics, the synthetic datasets' distributional behaviors are nearly identical to the real data distribution.

(a) RCGAN

(b) TimeGAN

(c) CWGAN

(d) RCWGAN
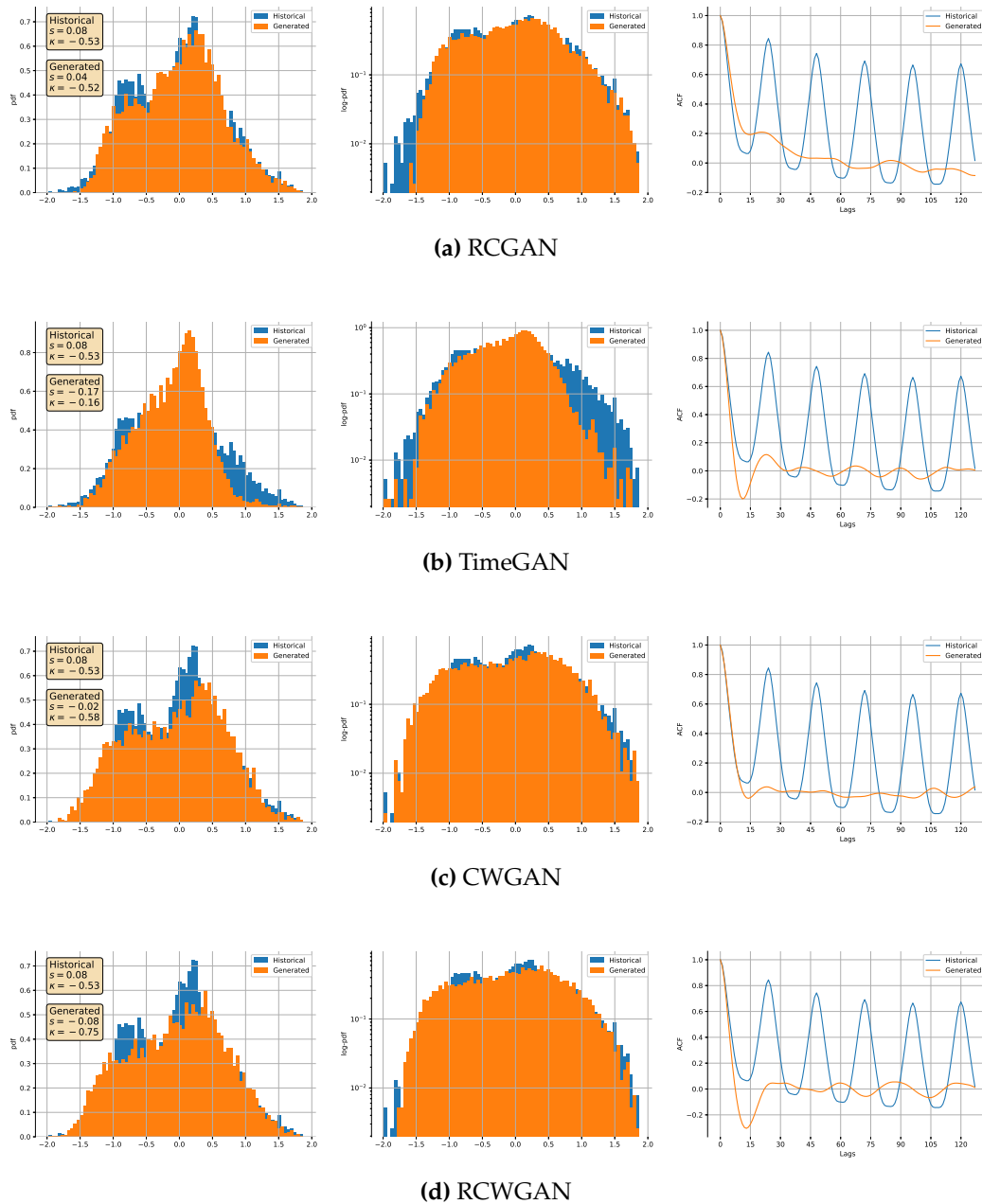
Figure 4. The generated and original load are compared in terms of their marginal distributions using a linear scale (1st column), log-plot (2nd column), and the auto-correlation fit with real load data

Figure 5 presents scenarios generated by the GANs models, namely RCGAN, TimeGAN, CWGAN, and RCWGAN. The figure presents 500 synthetic data generated by these models with gray curves. Also, it includes a randomly selected synthetic data realization for each GAN (orange line) to compare with the real data (blue line) to observe the similarity in behavior between the two. The results show that the synthetic data generated by the GANs are bounded above and below, and none of the generated data is exploding. However, the figure reveals that some GAN-generated data points may have larger maximum and minimum values than the real data. This feature is particularly interesting for risk management analysis, such as controlling whether the electricity provider can handle extreme electricity demand. Additionally, the maximum and minimum electricity consumption values generated by TimeGAN are closer to the real data than the other GANs. Finally, the highlighted path's behavior closely mimics the empirical dataset's behavior.
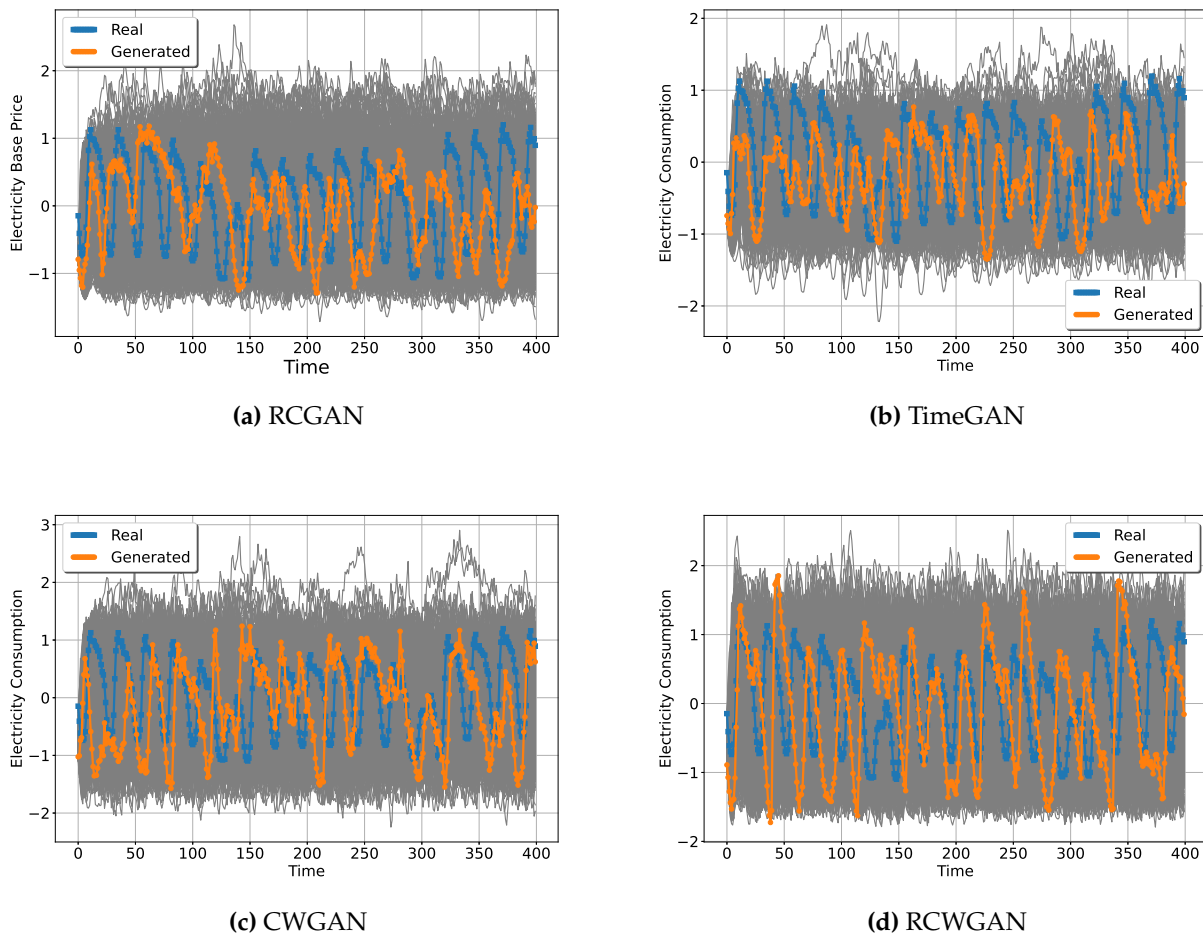
Figure 5. Synthetic load trajectories (orange and gray) generated by GANs and observed load (blue)

Table 3 provides key statistics for real and synthetic electricity consumption data generated by RCGAN, TimeGAN, CWGAN, and RCWGAN. The first column lists the statistical measure of interest: the mean, standard deviation, maximum, and minimum. The subsequent columns show the corresponding values for each real and GANs. The mean of the real historical electricity consumption data is -0.0453, while the mean of the synthetic data generated by RCWGAN is -0.04467. This suggests that RCWGAN can generate synthetic data that closely resembles the statistical properties of real data.

Similarly, the standard deviation of the synthetic load data generated by TimeGAN is lower than generated load data by other GANs, implying that TimeGAN produced synthetic load data that is less variable than the others. Although the table presents variations in the statistical properties, these variations are relatively small. Overall, the table provides a helpful summary of the statistical properties of the real and synthetic load data generated by the GANs. However, the table needs to provide more information to compare the success of the GANs. Furthermore, the table presents key statistics for a single synthetic load data. Hence, the comparison can change for other synthetic load data (see Figure 5).

5 Conclusion

In the realm of power grid regulations, precise simulation and prediction of load have become increasingly essential. Therefore, load modeling has been extensively researched using various methods, including regression-based and artificial intelligence (AI) modeling techniques. Over

Table 3. Key statistics of real and synthetic electricity consumption data

	Historical	RCGAN	TimeGAN	CWGAN	RCWGAN
Mean	-0.0453	-0.0043	-0.1320	-0.0436	-0.0467
Std	0.6773	0.6492	0.5111	0.7394	0.7225
Max	1.8637	2.0236	1.7280	2.4780	2.0411
Min	-1.9829	-1.5733	-1.9408	-1.9993	-1.7571

the last decade, AI models have gained significant attention due to their ability to model load without requiring detailed building and environmental parameters. Two primary approaches to AI modeling exist, deep learning and traditional machine learning, which rely heavily on real-time recorded load data. It is undeniable that time-dependent recorded load data serves as a critical source of information for energy market participants.

Having representative and diverse training load data is crucial to achieving good performance from AI models. However, obtaining such data can be challenging, costly, and time-consuming. In cases where there is insufficient load data or the sampling of load data deviates from the observed data distribution, the accuracy of model predictions can be significantly affected. As a result, energy suppliers may experience substantial trading losses, and energy sources may be overused, leading to more significant problems in the long run. Therefore, this paper proposes using GANs for synthetic load data generation. Specifically, it utilizes the RCGAN, TimeGAN, CWGAN, and RCWGAN models in real-world applications, achieving state-of-the-art results for synthetic load data generation. The findings suggest that GANs can be utilized to address data privacy concerns and enhance load modeling efficiency for grid modeling. As shown in Figure 4, the CWGAN and RCWGAN models perform relatively better than the other GANs, with the ability to capture values in the tails. In contrast, the TimeGAN model is unsuccessful in capturing tail values, resulting in a more peaked distribution than the real data distribution.

Future work can explore enhancing GAN efficiency by providing additional information during the training process. The study shows that GANs effectively generate synthetic data for load modeling, enabling risk management and analysis of various scenarios. The significance of this study lies in demonstrating the effectiveness of GANs for modeling electricity consumption patterns, allowing non-academic researchers and institutions to make informed decisions and develop strategies for energy-related challenges.

The results of our study demonstrate that the GANs proposed in this paper can effectively generate synthetic data for load modeling. Therefore, the generated data can be combined with existing empirical demand data to address risk management issues such as extreme demand tests, optimal timing of maintenance for wind turbines, energy efficiency assessments for buildings, and profitability analysis for demand- or time-dependent pricing strategies.

In essence, GANs offer a valuable solution for generating synthetic electricity consumption data, enabling non-academic researchers and institutions to gain insights, conduct simulations, and develop innovative approaches. This study opens up new possibilities for leveraging advanced AI techniques to improve energy management and contribute to a more sustainable and efficient energy future.

Declarations

List of abbreviations

Not applicable.

Ethical approval

The authors state that this research complies with ethical standards. This research does not involve either human participants or animals.

Consent for publication

Not applicable.

Conflicts of interest

The author confirms that there is no competing interest in this study.

Data availability statement

Data availability is not applicable to this article as no new data were created or analysed in this study.

Funding

Not applicable.

Author's contributions

The author has made substantial contributions to the conception, design of the work, the acquisition, analysis, interpretation of data, and the creation of new software used in the work.

Authors' information

Dr. Yilmaz holds an M.Sc. and Ph.D. degree in financial mathematics from the Institute of Applied Mathematics at Middle East Technical University. Currently, Dr. Yilmaz is a Post Doctoral researcher at RPTU Kaiserslautern, Germany.

References

[1] Moon, J., Jung, S., Park, S. and Hwang, E. Conditional tabular GAN-based two-stage data generation scheme for short-term load forecasting. *IEEE Access*, 8, 205327-205339, (2020). [CrossRef]

[2] Arjovsky, M., Chintala, S. and Bottou, L. Wasserstein generative adversarial networks. In Proceedings, *34th International Conference On Machine Learning (PMLR)*, (Vol. 70), pp. 214-223, (2017, July).

[3] Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V. and Courville, A. Improved training of wasserstein gans. In NeurIPS Proceedings, *Advances in Neural Information Processing Systems 30*, (2017).

[4] Yu, L., Zhang, W., Wang, J. and Yu, Y. SeqGAN: Sequence generative adversarial nets with policy gradient. In Proceedings, *Thirty-First AAAI Conference On Artificial Intelligence*, (Vol. 31, No. 1), pp. 2852-2858, (2017, February). [CrossRef]

[5] Zhu, J.Y., Park, T., Isola, P. and Efros, A.A. Unpaired image-to-image translation using cycle-consistent adversarial networks. In Proceedings, *IEEE International Conference on Computer Vision (ICCV)*, pp. 2223-2232, (2017, October). [CrossRef]

[6] Bendaoud, N., Farah, N. and Ben Ahmed, S. Comparing generative adversarial networks architectures for electricity demand forecasting. *Energy and Buildings*, 247, 111152, (2021). [CrossRef]

[7] Silva, V.L., Heaney, C.E., Li, Y. and Pain, C.C. Data assimilation predictive GAN (DA-PredGAN) applied to a spatio-temporal compartmental model in epidemiology. *Journal of Scientific Computing,* 94(1), 25, (2023). [CrossRef]

[8] Esteban, C., Hyland, S. and Rätsch, G. Real-valued (medical) time series generation with recurrent conditional gans. *ArXiv Prints, ArXiv:1706.02633,* (2017). [CrossRef]

[9] Yoon, J., Jarrett, D. and Van der Schaar, M. Time-series generative adversarial networks. *Advances In Neural Information Processing Systems*, 32, (2019).

[10] Yilmaz, B. and Korn, R. Synthetic demand data generation for individual electricity consumers: generative adversarial networks (GANs). *Energy and AI,* 9, 100161, (2022). [CrossRef]

[11] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S. et al. Generative adversarial nets. In NeurIPS Proceedings, *Advances in Neural Information Processing Systems 27,* (2014).

[12] Ramasinghe, S., Khan, S., Barnes, N. and Gould, S. Spectral-GANs for high-resolution 3D point-cloud generation. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS),* pp. 8169-8176, Las Vegas, NV, USA, (2020, October). [CrossRef]

[13] Wang, C., Wang, C., Xu, C. and Tao, D. Tag disentangled generative adversarial networks for object image re-rendering. In *International Joint Conference On Artificial Intelligence (IJCAI)*, Melbourne, Australia, (2017, August). [CrossRef]

[14] Tan, W.R., Chan, C.S., Aguirre, H.E. and Tanaka, K. ArtGAN: Artwork synthesis with conditional categorical GANs. In *2017 IEEE International Conference on Image Processing (ICIP)*, pp. 3760-3764, Beijing, China, (2017, September). [CrossRef]

[15] Merrigan, A. and Smeaton, A.F. Using a GAN to generate adversarial examples to facial image recognition. *ArXiv Preprint, ArXiv: 2111.15213,* (2021). [CrossRef]

[16] Engel, J., Agrawal, K., Chen, S., Gulrajani, I., Donahue, C. and Roberts, A. Gansynth: Adversarial neural audio synthesis. *ArXiv Preprint, ArXiv: 1902.08710,* (2019). [CrossRef]

[17] Donahue, C., McAuley, J. and Puckette, M. Adversarial audio synthesis. *ArXiv Preprint, ArXiv: 1802.04208,*(2018). [CrossRef]

[18] Young, T., Hazarika, D., Poria, S. and Cambria, E. Recent trends in deep learning based natural language processing. *IEEE Computational Intelligence Magazine,* 13(3), 55-75, (2018). [CrossRef]

[19] Englund, C., Aksoy, E.E., Alonso-Fernandez, F., Cooney, M.D., Pashami, S. and Astrand, B. AI perspectives in Smart Cities and Communities to enable road vehicle automation and smart traffic control. *Smart Cities,* 4(2), 783-802, (2021). [CrossRef]

[20] Wiese, M., Knobloch, R., Korn, R. and Kretschmer, P. Quant GANs: deep generation of financial time series. *Quantitative Finance*, 20(9), 1419-1440, (2020). [CrossRef]

[21] Mao, X., Li, Q., Xie, H., Lau, R.Y., Wang, Z. and Smolley, S.P. Least squares generative adversarial networks. In Proceedings, *IEEE International Conference on Computer Vision (ICCV)*, pp. 2794-2802, (2017, April).

[22] Mirza, M. and Osindero, S. Conditional generative adversarial nets. *ArXiv Preprint, ArXiv: 1411.1784,* (2014). [CrossRef]

[23] Li, J., Wang, X., Lin, Y., Sinha, A. and Wellman, M. Generating realistic stock market order streams. In Proceedings, *Conference on Artificial Intelligence (AAAI)*, (Vol. 34, No. 01), pp. 727-734, (2020, April). [CrossRef]

[24] Zhang, K., Zhong, G., Dong, J., Wang, S. and Wang, Y. Stock market prediction based on generative adversarial network. *Procedia Computer Science*, 147, 400-406, (2019). [CrossRef]

[25] Koshiyama, A., Firoozye, N. and Treleaven, P. Generative adversarial networks for financial trading strategies fine-tuning and combination. *Quantitative Finance,* 21(5), 797-813, (2021). [CrossRef]

[26] Chen, Y., Wang, Y., Kirschen, D. and Zhang, B. Model-free renewable scenario generation using generative adversarial networks. *IEEE Transactions on Power Systems,* 33(3), 3265-3275, (2018). [CrossRef]

[27] Li, J., Chen, Z., Cheng, L. and Liu, X. Energy data generation with wasserstein deep convolutional generative adversarial networks. *Energy*, 257, 124694, (2022). [CrossRef]

[28] Du, Z., Chen, K., Chen, S., He, J., Zhu, X. and Jin, X. Deep learning GAN-based data generation and fault diagnosis in the data center HVAC system. *Energy and Buildings*, 289, 113072, (2023). [CrossRef]

[29] Wang, Z. and Hong, T. Generating realistic building electrical load profiles through the generative adversarial network (GAN). *Energy and Buildings*, 224, 110299, (2020). [CrossRef]

[30] Perera, A.T.D., Khayatian, F., Eggimann, S., Orehounig, K. and Halgamuge, S. Quantifying the climate and human-system-driven uncertainties in energy planning by using GANs. *Applied Energy*, 328, 120169, (2022). [CrossRef]

[31] Dong, W., Chen, X. and Yang, Q. Data-driven scenario generation of renewable energy production based on controllable generative adversarial networks with interpretability. *Applied Energy*, 308, 118387, (2022). [CrossRef]

[32] Chen, Z., Li, J., Cheng, L. and Liu, X. Federated-WDCGAN: A federated smart meter data sharing framework for privacy preservation. *Applied Energy*, 334, 120711, (2023). [CrossRef]

[33] Ye, Y., Strong, M., Lou, Y., Faulkner, C.A., Zuo, W. and Upadhyaya, S. Evaluating performance of different generative adversarial networks for large-scale building power demand prediction. *Energy And Buildings,* 269, 112247, (2022). [CrossRef]

[34] Sözen, A. and Arcaklioglu, E. Prediction of net energy consumption based on economic indicators (GNP and GDP) in Turkey. *Energy Policy*, 35(10), 4981-4992, (2007). [CrossRef]

[35] Hamzaçebi, C. Forecasting of Turkey's net electricity energy consumption on sectoral bases. *Energy Policy*, 35(3), 2009-2016, (2007). [CrossRef]

[36] Kankal, M., Akpınar, A., Kömürcü, M.İ. and Özşahin, T.Ş. Modeling and forecasting of Turkey's energy consumption using socio-economic and demographic variables. *Applied Energy*, 88(5), 1927-1939, (2011). [CrossRef]

[37] Es, H.A., Kalender Öksüz, F.Y. and Hamzacebi, C. Forecasting the net energy demand of Turkey by artificial neural networks. *Journal of The Faculty of Engineering and Architecture of Gazi University*, 29(3), (2014).

[38] Ağbulut, Ü. Forecasting of transportation-related energy demand and CO_2 emissions in Turkey with different machine learning algorithms. *Sustainable Production and Consumption*, 29, 141-157, (2022). [CrossRef]

[39] Yasin Çodur, M. and Ünal, A. An estimation of transport energy demand in Turkey via artificial neural networks. *Promet-Traffic & Transportation*, 31(2), 151-161, (2019). [CrossRef]

[40] Yilmaz, B. Understanding the mathematical background of generative adversarial neural networks (GANs). *Available At SSRN 3981773*, (2021). [CrossRef]

[41] Wang, Y. A mathematical introduction to generative adversarial nets (GAN). *ArXiv Preprint, ArXiv: 2009.00169,* (2020). [CrossRef]

[42] Ni, H., Szpruch, L., Wiese, M., Liao, S. and Xiao, B. Conditional sig-wasserstein GANs for time series generation. *ArXiv Preprint, ArXiv: 2006.05421,* (2020). [CrossRef]

[43] Yukseltan, E., Kok, A., Yucekaya, A., Bilge, A., Aktunc, E.A. and Hekimoglu, M. The impact of the COVID-19 pandemic and behavioral restrictions on electricity consumption and the daily demand curve in Turkey. *Utilities Policy*, 76, 101359, (2022). [CrossRef]

[44] Kingma, D.P. and Ba, J. Adam: A method for stochastic optimization. *ArXiv Preprint, ArXiv: 1412.6980,* (2017). [CrossRef]

[45] Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B. and Hochreiter, S. GANs trained by a two time-scale update rule converge to a local nash equilibrium. In NeurIPS Proceedings, *Advances in Neural Information Processing Systems 30,* (2017).

[46] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G. et al. Pytorch: An imperative style, high-performance deep learning library. In NeurIPS Proceedings, *Advances in Neural Information Processing Systems 32,* (2019).

Mathematical Modelling and Numerical Simulation with Applications (MMNSA)
(https://dergipark.org.tr/en/pub/mmnsa)