PV Fleet Modeling via Smooth Periodic Gaussian Copula

Mehmet G. Ogut[1], Bennet Meyers[2], and Stephen P. Boyd[1]
[1] Stanford University, Stanford, CA, 94305, USA
[2] SLAC National Accelerator Laboratory, Menlo Park, CA, 94025, USA

2023 IEEE 50th Photovoltaic Specialists Conference (PVSC) | 978-1-6654-6059-0/23/$31.00 ©2023 IEEE | DOI: 10.1109/PVSC48320.2023.10359991

***Abstract*—We present a method for jointly modeling power generation from a fleet of photovoltaic (PV) systems. We propose a white-box method that finds a function that invertibly maps vector time-series data to independent and identically distributed standard normal variables. The proposed method, based on a novel approach for fitting a smooth, periodic copula transform to data, captures many aspects of the data such as diurnal variation in the distribution of power output, dependencies among different PV systems, and dependencies across time. It consists of interpretable steps and is scalable to many systems. The resulting joint probability model of PV fleet output across systems and time can be used to generate synthetic data, impute missing data, perform anomaly detection, and make forecasts. In this paper, we explain the method and demonstrate these applications.**

***Index Terms*—photovoltaic systems, photovoltaic fleet modeling, distributed power generation, power generation planning, forecasting, convex optimization, copula method, probability distributions, forecast uncertainty**

I. Introduction

Modeling the power output of a fleet of photovoltaic (PV) systems is of great importance for digital operations and maintenance in the PV sector, which is now a multi-billion dollar industry [1]. Applications include predicting power production, detecting anomalies, and making informed decisions about when and where to send workers to service a site. In recent years, there has been a significant increase in the deployment of PV systems, making it necessary to develop scalable models that can handle thousands of systems simultaneously, while staying robust to real-world data challenges, such as the ability to handle missing data.

In this paper we propose a method to estimate the joint probability distribution of the power outputs of a fleet of PV systems, modeling all relevant correlations in the data—across individual PV systems and across time. As a copula method, we first develop a novel set of nonlinear marginal transforms that map the power from each system to a scalar Gaussian, and then develop a set of linear transformations that model the marginally transformed data as a large joint Gaussian distribution. We interact with that model to carry out various applications including synthetic data generation, data imputation, anomaly detection, and forecasting.

II. Prior work

a) Fleet models: Modeling PV systems for operations and maintenance (O&M) purposes based on measured data has a long history [2]–[7]. These techniques focus on predicting either the maximum power point or the full current-voltage relationship of a PV system under a given set of environmental conditions. O&M tasks are then carried out using the model. Fault detection is performed by comparing actual system power generation to the predicted power from the model. Forecasting future PV system power output is done by running the models on predicted weather trends, such as those generated by NOAA [8]. 'Fleet modeling' is the practice of constructing an independent, bespoke model for each system in a PV fleet and is quite labor intensive. More recently, researchers have attempted to reduce the effort of fleet modeling using systematic approaches such as learning algorithms and machine inference [9]. These approaches tend to be more task dependent, *e.g.*, focusing on the task of anomaly detection [10]–[12] or forecasting [13]–[19]. While these methods reduce the human effort to create a PV system model, they do not provide *joint* models of system behavior in a fleet. Several recent papers have explored models to predict aggregate quantities of fleets, such as temporal variability and maximum feed-in power [20]–[23].

b) Copula models and Gaussianization: Our proposed method draws on previous work on both copula models and Gaussianization methods.

Copulas are tools for modeling dependence of several random variables, first proposed by Abe Sklar in 1959 [24] and recently translated into English in [25]. The basic idea is to apply a nonlinear invertible mapping to each component of a random variable so it has some standard distribution such as uniform or Gaussian, and then model the dependence of these transformed variables [26]–[28]. Copulas have been applied in many domains [29]–[33], including PV data analysis [34]. Copula models may be based on theoretical constructions (*e.g.*, the multivariate Gaussian copula) or may be learned directly from data [27]. The *quantile transform* is a typical choice for data-driven models of the marginal distributions, and various options exist [35], [36]. Our method includes an autoregressive component in the copula model, an approach that has been explored by other authors [37].

An alternative approach to modeling multivariate joint distributions is *Gaussianization*, also called 'normalizing flows'. These methods seek an invertible mapping under which the transformed variable has a standard (jointly) Gaussian distribution [38]–[40]. The transformations are typically built in steps, so the transformation is a composition of multiple

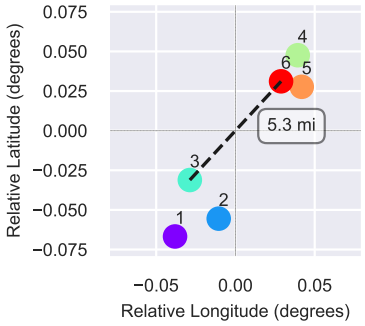

Fig. 1: Relative location of the six PV systems.

transformations, with the distribution of mapped variables getting closer to Gaussian as more layers or steps are added. We can think of a Gaussian copula as a simple first step in such a normalizing flow. While a normalization method can in principle model any probability distribution, Gaussian copula models cannot. On the other hand a Gaussian copula model makes several operations such as conditioning on some known values very easy, involving just basic linear algebra. (These computions can be done for more complex normalizing flows, but they are much more involved, *e.g.*, requiring Monte Carlo or other sampling type methods.)

c) Modeling via convex optimization: Our method relies on convex optimization in every step. This guarantees efficient algorithms for finding global solutions [41], and mature tools exist to easily specify convex optimization problems in code [42], [43]. Many traditional statistical models rely on convex optimization for fitting such as regression [44, Chap. 12], auto-regressive (AR) models [44, Ch. 13], and fitting Gaussian distributions to data [41, §3.5], among many others. Our method is inspired by recent work on the trade-off of fit versus roughness in stratified Gaussian models [45], [46], as well as work on convex optimization based signal decomposition [47].

III. Data

We will illustrate our method on PV fleet data provided by SunPower Corporation under a nondisclosure agreement. We select six PV systems located in Southern California, with three grouped in Santa Ana, CA and three grouped in the hills to the east in Tustin, CA. The relative locations are shown in figure 1. This choice of system locations was intentional, as we wanted to verify that our model captures similarity of power profiles for nearby systems.

The data consist of 15-minute (average) power values (in kW) for each of the six systems, recorded from 3/1/2017 to 3/31/2017. Figure 2 depicts the power output of the six systems over the three day period between 3/4/2017 and 3/6/2017. At night the power output is zero; during daylight hours, we see different types of power profiles. On 3/6/2017 we see clear-sky behavior, characterized by a smooth increase until noon followed by a gradual decrease until evening. On 3/5/2017, however, we see the power generation curves with multiple dips and peaks throughout the day, which can be attributed to weather factors such as passing clouds. The maximum power output of the systems varies, with system 4 peaking at around 9 kW, and the other systems peaking at around 2 kW.

We denote the data as $y_t \in \mathbf{R}^d$, with $d = 6$, and the time index running from $t = 1$ to $t = T = 2976$. This particular data set does not have any missing data. However, our method gracefully handles missing values, and indeed, relies on this ability to choose hyper-parameters by cross-validation.

We also use data for the following 2 weeks, from 4/1/2017 to 4/14/2017 as our test set for validating our models and applications. This data was not used to fit our model. The test data set has index running from $t = 1$ to $T^{\text{test}} = 1344$.

IV. Method

We propose a method for fitting the given data $y_1, \ldots, y_T$ to a smooth 24-hour-periodic stochastic process. We apply a sequence of three invertible transformations so that the transformed data is approximately a standard Gaussian. These transformations, which respect periodicity and are constructed to vary smoothly across time, are applied in the three steps shown in figure 3. First we use a smooth periodic nonlinear transform to make the data approximately marginally Gaussian. This allows us to model the changing distribution over a day, in addition to the differing maximum values seen across systems. In the second step we use an autoregressive (AR) model to account for dependencies across time. This results in a residual that is approximately uncorrelated across time. Finally, we fit a smooth periodic Gaussian distribution to the residual of the AR model; from this we can whiten the residual so that it is approximately a standard Gaussian. In the language of copula modeling, our first step is our marginal transformation and the final two steps constitute our copula (*i.e.*, 'linking') function.

A. Fitting a smooth periodic model

Here we describe the general technique, used in steps 1 and 3 of our method, for fitting a smooth P-periodic parameter, given by $\theta_1, \ldots, \theta_T \in \Theta \subseteq \mathbf{R}^m$ to some data, where Θ is a convex set of allowed parameter values. We will use a Fourier series with K harmonics to represent θ,

$$\theta_t = \sum_{k=0}^{K} \left(\cos\left(\frac{2\pi k t}{P}\right) \alpha_k + \sin\left(\frac{2\pi k t}{P}\right) \beta_k \right), \qquad (1)$$

for $t = 1, \ldots, P$, where $\alpha_k, \beta_k \in \mathbf{R}^m$ are the (vector) coefficients that define θ.

We take as a measure of smoothness the Dirichlet energy,

$$\mathcal{D} = \frac{(2\pi)^2}{P} \sum_{k=1}^{K} k^2 \left(\|\alpha_k\|_2^2 + \|\beta_k\|_2^2 \right).$$

The loss function has the form

$$\mathcal{L} = \sum_{t=1}^{T} \ell_t(\theta_t),$$

978-1-6654-6059-0/23/$31.00 ©2023 IEEE

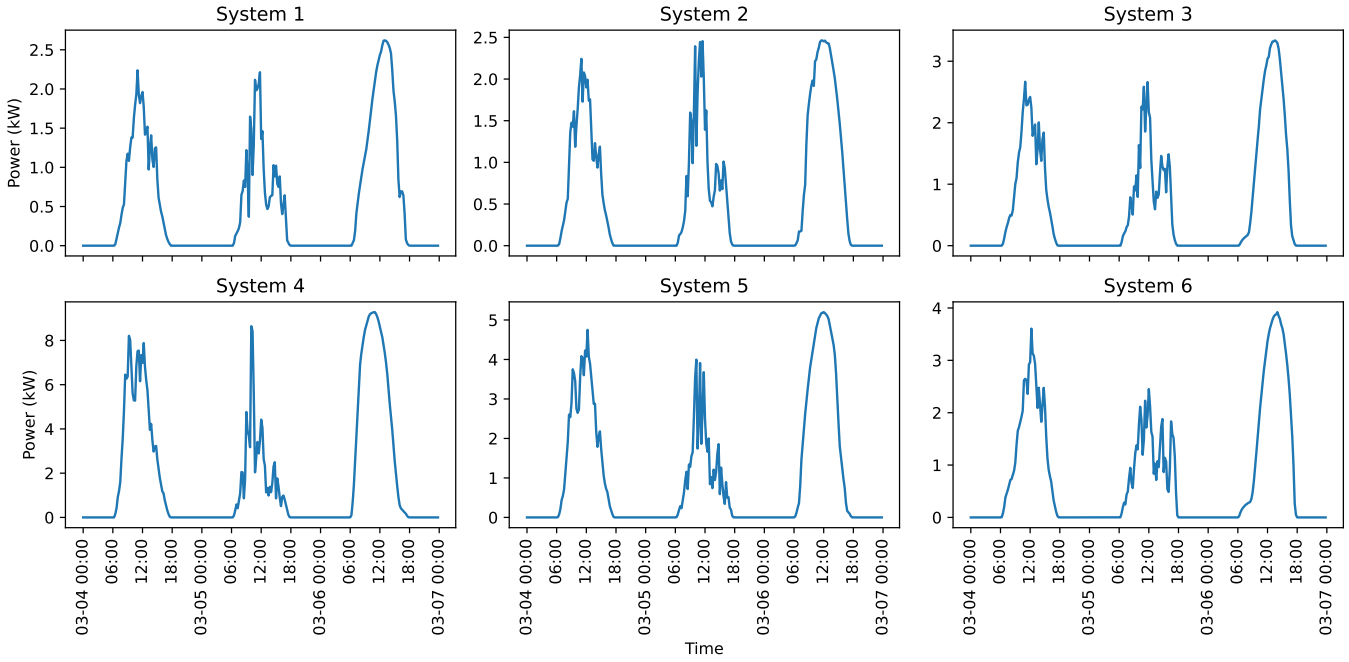

Fig. 2: 15 minute power output data for six PV systems from 3/4/2017 to 3/6/2017.

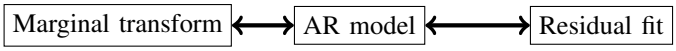

Fig. 3: Three invertible tranformations used in the proposed method.

where $\ell_t : \Theta \to \mathbf{R}$ is a convex loss function that depends on some data at time t. (If some data are missing, then the sum is only over t for which the data are available.)

Our generic fitting method takes θ as a solution of the convex optimization problem

$$\begin{array}{ll} \text{minimize} & \mathcal{L} + \lambda \mathcal{D}, \\ \text{subject to} & \theta_t \in \Theta, \quad t = 1, \ldots, P, \end{array} \tag{2}$$

where $\lambda > 0$ is the smoothing regularizer hyper-parameter. The variables are the m-vectors $\alpha_0, \ldots, \alpha_K$ and $\beta_1, \ldots, \beta_K$. This is a convex optimization problem, and readily solved.

This generic fitting method contains the hyper-parameter λ (and possibly others), but good values of these can be found automatically using cross-validation [48, §7.10], [44, §13.2], so the method is essentially hyper-parameter free and automatic.

Two steps of our method solve a different instance of the problem (2) with λ chosen using cross-validation. This method of fitting a smooth periodic parameter is a special case of a Laplacian regularized stratified model [45], [46] with the underlying graph a cycle representing the periodicity. It can also be thought of as a signal decomposition problem [47].

B. Marginal transforms

In this first step, we seek continuous increasing functions $\varphi_{t,i} : \mathbf{R} \to \mathbf{R}$. We allow these functions to change with t but enforce that they are periodic and smooth. Our goal is for the transformed values,

$$x_{t,i} = \varphi_{t,i}(y_{t,i}), \qquad \text{for } t = 1, \ldots, T,$$

to have an approximately Gaussian (marginal) distribution for each $i = 1, \ldots, d$. Typical quantile transforms are static; but our method defines a transformation that changes smoothly in time and is periodic.

We carry out this step for each component of the original data, so the method described in this section is carried out separately for each $i = 1, \ldots, d$. To lighten the notation in this section, we drop the component index i, and consider the original data y_t to be scalar.

Our first step is to estimate a set of quantiles $0 \leq \eta_1 < \cdots < \eta_r \leq 1$ of the data. By default we take these to be 2nd percentile, the 98th percentile, and the 10 deciles, so $r = 11$ and

$$\eta = (0.02, 0.10, 0.20, \ldots, 0.80, 0.90, 0.98),$$

but our method is general and any other choice of quantiles could be used. We denote the estimated η_i quantile of y_t as $q_{t,i}$, $t = 1, \ldots, T$, $i = 1, \ldots, r$. We assume these are P-periodic and smooth.

To estimate these quantiles from the data we use standard quantile regression [49], [50], which relies on the so-called pinball loss, defined as

$$\ell^{\text{pin}}(u; \eta) = \max\{(1-\tau)u, \tau u\} = (\tau - 1/2)|u| + (1/2)u,$$

for quantile $\tau \in [0, 1]$. We take our loss function to be

$$\ell(q_t) = \ell^{\text{pin}}(q_{t,1} - y_t; \eta_1) + \cdots + \ell^{\text{pin}}(q_{t,r} - y_t; \eta_r), \tag{3}$$

978-1-6654-6059-0/23/$31.00 ©2023 IEEE

the sum of the pinball losses associated with each of our r quantiles. To estimate the quantiles we solve the generic problem (2), with loss function (3), and constraint set

$$\Theta = \{q \mid q_1 \leq \cdots \leq q_r\},$$

which enforces that the quantiles are consistent. This simultaneously estimates the r quantiles of y_t for each t, with the quantiles being smooth and periodic, and always satisfiying the consistency constraint. The hyper-parameter λ, which controls how smooth the quantile estimates are, can be chosen automatically via cross-validation.

Given these periodic smooth quantile estimates, we construct nonlinear mappings $\varphi_t : \mathbf{R} \to \mathbf{R}$ as continuous piecewise linear functions, with knot-points given by the quantiles, and values at those points given by the associated value of a standard scalar Gaussian for the same quantiles, *i.e.*,

$$\varphi_t(q_{t,j}) = \Phi^{-1}(\eta_j), \quad j = 1, \ldots, r, \tag{4}$$

where Φ is the cumulative distribution function (CDF) of a standard Gaussian. This gives the marginally transformed data $x_{t,i} = \varphi_t(y_{t,i})$.

C. Autoregressive model

The time series $x_1, \ldots, x_T$ has entries with approximately standard Gaussian marginal distribution, but there are dependencies between the components, as well as across time. Our next step is to handle the dependency across time. We fit a vector autoregressive (AR) model to the marginally Gaussianized data x_t. The model is

$$x_t = A_1 x_{t-1} + \cdots + A_M x_{t-M} + v_t, \tag{5}$$

where v_t is a process noise or residual, M is the memory of the AR model, and $A_1, \ldots, A_M \in \mathbf{R}^{d \times d}$ are the coefficients. We could fit these AR coefficients as smooth and periodic, but we have found that a constant AR model does just as well as a more complex time-varying one.

We fit these coefficients by minimizing the mean-squared error, the average of

$$\ell_t = \|x_t - A_1 x_{t-1} + \cdots + A_M x_{t-M}\|_2^2$$

over those entries where all x_t are known. We add ridge regularization to this average loss,

$$\lambda^{\text{ridge}} \left(\|A_1\|_F^2 + \cdots + \|A_M\|_F^2 \right),$$

where $\lambda^{\text{ridge}} > 0$ is a hyper-parameter that scales the regularization, and $\|\cdot\|_F^2$ denotes the square of the Frobenius norm, *i.e.*, the sum of squares of entries. The hyperparameter λ^{ridge} can be chosen by cross-validation.

We denote the AR residual as

$$v_t = x_t - (A_1 x_{t-1} + \cdots + A_M x_{t-M}),$$

defined when $x_t, \ldots, x_{t-M}$ are all known. Note that in the special case $M = 0$, which corresponds to the model that x_t are approximately uncorrelated for different t, the residual reduces to x_t.

D. Smooth periodic residual fit

Our last step is fit a smoothly varying periodic Gaussian distribution to the residual v_t, $v_t \sim \mathcal{N}(\mu_t, \Sigma_t)$, where we assume that v_s and v_t are independent for $s \neq t$. We model $v_t \in \mathbf{R}^d$ as smooth periodic Gaussian,

$$v_t \sim \mathcal{N}(\mu_t, \Sigma_t), \quad t = 1, \ldots, T \tag{6}$$

where Σ_t and μ_t are smooth and periodic. We expect μ_t to be small.

Our loss ℓ_t will be the negative log-likelihood of the Gaussian model (6), which is

$$\ell_t = \frac{d}{2}\log(2\pi) - \sum_{j=1}^{d} \log\left(\mathbf{diag}\,(L_t)_j\right) + \frac{1}{2}\|L_t^T v_t - \nu_t\|_2^2,$$

with variables $L_t \in \mathbf{R}^{d \times d}$ and $\nu_t \in \mathbf{R}^d$. Here, L_t is the Cholesky factor of Σ_t^{-1} and $\nu_t = L_t^{-T}\mu_t$. This change of variables makes the loss a convex function. We can recover Σ_t and μ_t as

$$\Sigma_t = (L_t L_t^T)^{-1}, \qquad \mu_t = L_t^{-1}\nu_t.$$

once we solve the optimization problem to find L_t and ν_t. Here too the smoothness hyper-parameter λ is found by cross-validation.

Our final whitened signal is given by

$$z_t = \Sigma_t^{-1/2}(x_t - \mu_t) = L_t^T x_t - \nu_t,$$

defined when x_t is. According to our model, these are independent identically distributed (IID) with $z_t \sim \mathcal{N}(0, I)$.

E. The whole model

From our first two steps, we see that our model of y_t is a stationary periodic Gaussian process x_t, mapped entrywise through a smoothly periodic transformation. Using all three steps, we interpret it as IID Gaussians z_t, passed through an AR filter to obtain y_t, and then mapped entrywise.

Such a model allows us to carry out several operations. We can generate samples from the model. We can evaluate the density at a sequence y_t. We can condition on a set of known values of some of the components, as well as computing conditional marginal quantiles for each unknown entry. These allow us to carry out imputation, *i.e.*, guessing missing values, by evaluating the conditional median of a missing entry given the known ones. (We also can get error bars, *e.g.*, the 10th and 90th conditional marginal quantiles.) We can also do anomaly detection, where we detect known entries that do not fit the model. To do this we compute the conditional quantile of each known entry, given the other known entries but not that particular value; conditional quantile values that are either near zero or one are then flagged as suspicious.

These operations (and others) can be carried out for many types of statistical models, for example using Monte Carlo sampling. But due to the specific structure of our model, we can carry them out using simple linear algebra, which makes the operations fast and reliable. Details of how we implement these operations will be given in a forthcoming paper.

978-1-6654-6059-0/23/$31.00 ©2023 IEEE

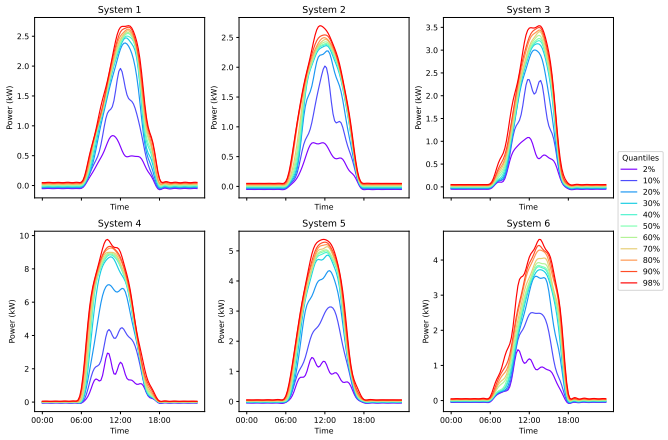

Fig. 4: Smooth periodic quantiles.

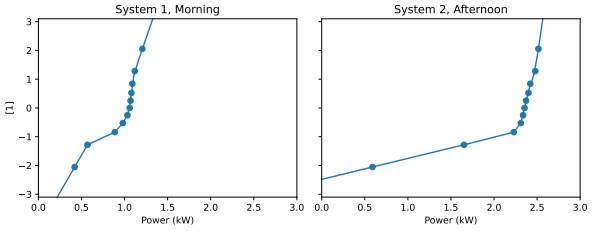

Fig. 5: Time aware copula transform.

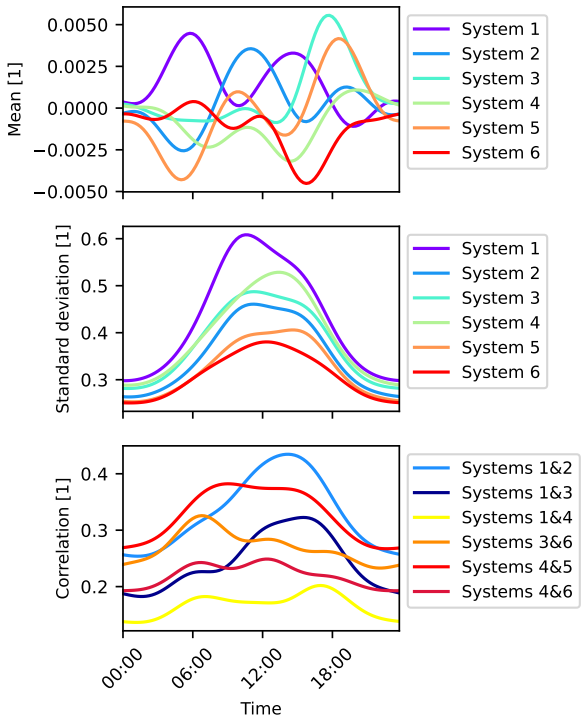

Fig. 6: Means, standard deviations, and selected correlations of AR residuals.

V. Results

Here we show the results of our modeling method on the PV data described above, using default parameters. Estimated quantiles for each system are shown in figure 4. Note that the 98th percentile serves as an effective statistical clear-sky model. The estimated quantiles collapse to zero at night as expected. The spread between the upper quantiles is narrower than that of the lower quantiles, especially around noon. Quantiles for each system exhibit distinct characteristics, with the systems physically near each other showing similar shaped quantiles.

A few samples of the associated piecewise linear copula transforms are shown in figure 5. We see that the same power value is mapped to different points based on the time of day and the system. This shows how our time-aware copula transform adapts to both the time of day and the unique characteristics of different systems as opposed to a standard fixed copula transform, which does not.

We use AR memory $M = 3$, with coefficients. We observe several interesting phenomena in these coefficient matrices. First, the entries of A_1 are generally bigger than those of A_2 and A_3, showing that the previous period plays a larger role in predicting the current values than the previous two values. We also see that the diagonal entries are generally larger than the off-diagonal ones, meaning that the previous values for each system play a larger role in predicting the current value than the previous values of the other systems. However, the many non-zero off-diagonal elements in the coefficient matrices show that the predictions for each system do depend on the previous values of the other systems.

Finally we fit a smoothly varying periodic Gaussian distribution to the residuals of the AR model, shown in figure 6. The top plot shows the means, which are indeed small, as expected. The middle plot shows the standard deviations of the residuals. These are smaller than one, which is approximately the standard deviation of the entries of x_t, which means we are able to predict the current values using previous values better than simply guessing $x_t = 0$, *i.e.*, treating them as uncorrelated across time. We can see that the residual standard deviations vary considerably across systems and time of day. Roughly speaking, the residual of system 1 has almost twice the standard variation of the residual of system 6. We also see that the residual standard deviation is smaller at noon than in the morning and afternoon. The bottom plot shows the correlations of selected pairs of residuals. These correlations are generally around 30%, but we can see that systems that are physically near each other are more highly correlated. We can also see variation of the correlation over the day.

VI. Applications

A. Generating simulated data

We generate simulated data from our model by simulating data from the periodic Gaussian stochastic process, and then applying the inverse nonlinearities $\phi_{t,i}^{-1}$ to the entries of these samples. Figure 7 shows two simulations of fake data for

978-1-6654-6059-0/23/$31.00 ©2023 IEEE

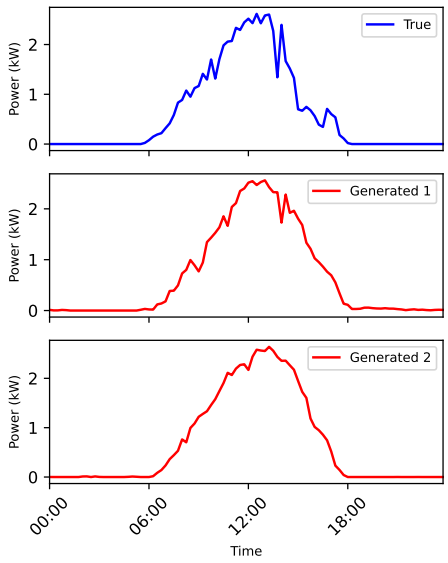

Fig. 7: *Top.* System 1 on 4/7/2017. *Middle and bottom.* Two simulations from our model.

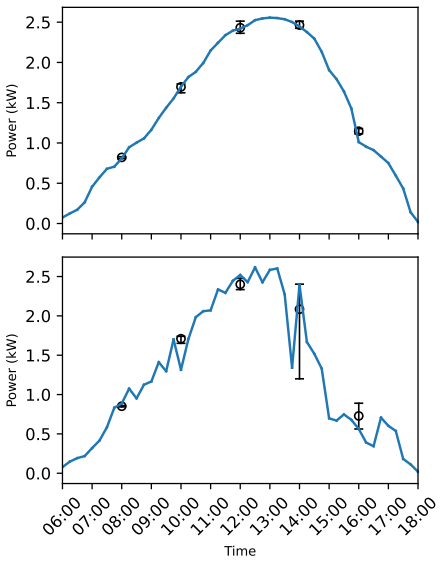

Fig. 8: 10th, 50th, and 90th marginal conditional quantiles at five times, conditioned on values of all systems at other times. *Top.* Clear day. *Bottom.* Cloudy day.

system 1, with the actual data for the specific day 4/7/2017 shown at top for reference. They appear quite similar.

B. Conditional marginal quantiles

We can compute the marginal quantiles of any entry, conditioned on all other known entries, in time and across systems. When the entry is unknown, this gives us a sophisticated method for imputing or guessing what the missing value might have been. We can use the conditional marginal median (50th percentile) as the imputed value, with the 10th and 90th percentiles defining an uncertainty interval. Figure 8 shows the marginal conditional quantiles for system 1 at 5 times on two days, one clear and one partially cloudy. We observe that the model correctly adapts the uncertainty bounds to the weather, with tighter bounds on clear days. Additionally, we note that the uncertainty bounds are asymmetric, with decreases in output (say, due to clouds) more likely than increases. The predictions themselves, shown as the circle representing the conditional median, are good.

C. Anomaly detection

We can use marginal conditional quantiles to identify anomalous entries in our data. To estimate whether a given known entry is an anomaly we pretend that it is unknown, compute its conditional marginal CDF given all other known values, and evaluate it at the known value. We can flag an entry as anomalous if this quantile value is less than ϵ or more than $1-\epsilon$, where ϵ is a threshold value such as 10^{-2}. With this threshold, we would expect a false positive rate around 2ϵ.

To illustrate this, we consider system 2 on 4/1/2017. We introduce synthetic anomalies by perturbing the power values at 8:30, 10:00, 11:30, 13:00, 14:15 and 15:30, by randomly increasing or decreasing the true values by 15%. Table I shows

TABLE I: Anomaly detection example.

Time	True value	Perturbed value	Conditional quantile
08:30	0.9919	0.8431	0.0001
10:00	1.7280	1.9872	0.9999
11:30	2.3633	2.7178	0.9999
13:00	2.5890	2.9773	0.9999
14:15	2.4294	2.0650	0.0001
15:30	1.6576	1.4090	0.0013

true values, perturbed values and conditional marginal quantiles of perturbed values, clipped to the range $[0.0001, 0.9999]$.

With threshold $\epsilon = 0.01$, we detect all of the artificial anomalies. We also have three false positives, *i.e.*, times when a true value is flagged as an anomaly. The conditional marginal quantiles for this day are shown in figure 9. The vertical axis shows $\min\{q, 1-q\}$, with the threshold $\epsilon = 0.01$ shown as the darker horizontal line. True negatives are shown as blue circles, and true positives are shown as blue squares. False positives are shown as orange circles. The three false positives are all cases where the true power was low compared to our predicted median. This is not suprising; clouds can easily reduce power output unexpectedly by 15% or more.

D. Forecast

Here we forecast the values of system 2 from 13:15 on, using data from all systems up through 13:00. We show the forecast, which is the median or 50th conditional marginal quantile, along with the conditional marginal 10th and 90th quantiles, which give us confidence bands for the forecast values. This is illustrated in figure 10 on the clear day 4/9/2017 and the cloudy day 4/6/2017. Our forecast on the clear day is very good, with tight uncertainty bands. Our forecast on the

978-1-6654-6059-0/23/$31.00 ©2023 IEEE

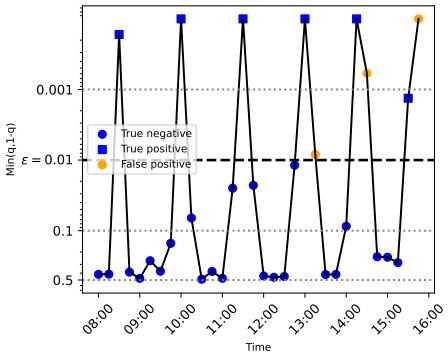

Fig. 9: Marginal conditional quantiles for system 2 on 4/1/2017, with six artificial anomalies added.

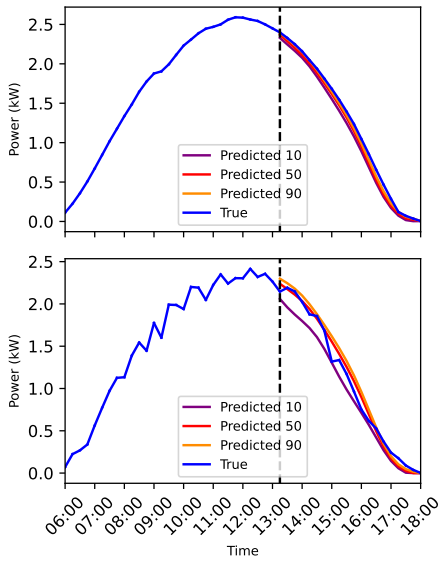

Fig. 10: Forecast for system 2 on the clear day 4/9/2017, and the cloudy day 4/6/2017. We forecast from 13:15 on, given values for all systems up through 13:00.

cloudy day is reasonable, but with much wider uncertainty bands.

In addition to forecasting marginal quantiles of a single system, we can generate joint forecasts using all systems. We illustrate this in figure 11, where we forecast the values of system 2 from 13:15 on, using data from all systems up through 13:00. We show three different forecasts, sampled from the full joint conditional distribution. We see that the forecasts agree with the marginal forecasts, in the sense that generated instances are within the 10–90% confidence bands of the marginal forecasts. We also see that the forecasts are reasonable, since we observe that for cloudy days, the forecasts are more volatile than for clear days with the latter having a more stable, smooth behavior.

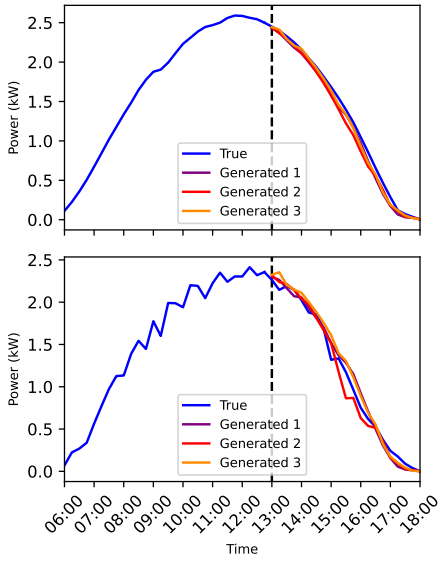

Fig. 11: Forecast for system 2 on the clear day 4/9/2017, and the cloudy day 4/6/2017. We forecast from 13:15 on, given values for all systems up through 13:00.

VII. Conclusions

We presented a novel approach to modeling and analyzing fleets of PV systems based on a smooth periodic Gaussian copula transform, and illustrated some of its applications. While we have demonstrated the method on a small example, it can scale gracefully to much larger problems; details will be given in a forthcoming paper.

Acknowledgment

This material is based on work supported by the U.S. Department of Energy's Office of Energy Efficiency and Renewable Energy (EERE) under the Solar Energy Technologies Office Award Number 38529. Stephen Boyd's work was funded in part by the AI Chip Center for Emerging Smart Systems (ACCESS).

References

[1] L. G. da Fonseca, "The state of digital O&M for the solar market," *Greentech Media*, October 2019, accessed May 24 2023. [Online]. Available: https://www.greentechmedia.com/articles/read/the-state-of-digital-om-for-the-solar-market1

[2] D. L. King, J. A. Kratochvil, and W. E. Boyson, "Measuring solar spectral and angle-of-incidence effects on photovoltaic modules and solar irradiance sensors," *IEEE 26th Photovoltaic Specialist Conference*, pp. 1113–1116, 1997.

[3] ——, "Temperature coefficients for PV modules and arrays: measurement methods, difficulties, and results," *IEEE 26th Photovoltaic Specialist Conference*, pp. 1183–1186, 1997.

[4] D. L. King, W. E. Boyson, and J. A. Kratochvil, "Photovoltaic array performance model." *Sandia Report No. 2004-3535*, vol. 8, pp. 1–19, August 2004. [Online]. Available: https://www.osti.gov/servlets/purl/919131/

[5] B. Marion, J. Adelstein, K. Boyle, H. Hayden, B. Hammond, T. Fletcher, B. Canada, D. Narang, A. Kimber, L. Mitchell, G. Rich, and T. Townsend, "Performance parameters for grid-connected PV systems," *IEEE 31st Photovoltaic Specialist Conference*, pp. 1601–1606, 2005.

[6] W. D. Soto, S. A. Klein, and W. A. Beckman, "Improvement and validation of a model for photovoltaic array performance," *Solar Energy*, vol. 80, pp. 78–88, January 2006.

[7] W. F. Holmgren, R. W. Andrews, A. T. Lorenzo, and J. S. Stein, "PVLIB Python 2015," *IEEE 42nd Photovoltaic Specialist Conference*, pp. 1–5, June 2015.

[8] R. G. Miller, "GEM: A statistical weather forecasting procedure," *NOAA Technical Report NWS-28*, November 1981. [Online]. Available: https://www.weather.gov/media/owp/oh/hdsc/docs/TR28.pdf

[9] S. Quintarelli, "Let's forget the term AI. let's call them systematic approaches to learning algorithms and machine inferences (SALAMI)," *Quinta's weblog*, November 2019, accessed May 10 2023. [Online]. Available: https://blog.quintarelli.it/2019/11/lets-forget-the-term-ai-lets-call-them-systematic-approaches-to-learning-algorithms-and-machine-inferences-salami/

[10] Y. Zhao, Q. Liu, D. Li, D. Kang, Q. Lv, and L. Shang, "Hierarchical anomaly detection and multimodal classification in large-scale photovoltaic systems," *IEEE Transactions on Sustainable Energy*, vol. 10, pp. 1351–1361, July 2019.

[11] F. Aziz, A. U. Haq, S. Ahmad, Y. Mahmoud, M. Jalal, and U. Ali, "A novel convolutional neural network-based approach for fault classification in photovoltaic arrays," *IEEE Access*, vol. 8, pp. 41 889–41 904, 2020.

[12] V. Veerasamy, N. I. A. Wahab, M. L. Othman, S. Padmanaban, K. Sekar, R. Ramachandran, H. Hizam, A. Vinayagam, and M. Z. Islam, "LSTM recurrent neural network classifier for high impedance fault detection in solar PV integrated power system," *IEEE Access*, vol. 9, pp. 32 672–32 687, 2021.

[13] P. Li, K. Zhou, X. Lu, and S. Yang, "A hybrid deep learning model for short-term PV power forecasting," *Applied Energy*, vol. 259, p. 114216, February 2020.

[14] Q. Huang and S. Wei, "Improved quantile convolutional neural network with two-stage training for daily-ahead probabilistic forecasting of photovoltaic power," *Energy Conversion and Management*, vol. 220, p. 113085, September 2020.

[15] S. Ding, R. Li, and Z. Tao, "A novel adaptive discrete grey model with time-varying parameters for long-term photovoltaic power generation forecasting," *Energy Conversion and Management*, vol. 227, p. 113644, January 2021.

[16] A. A. du Plessis, J. M. Strauss, and A. J. Rix, "Short-term solar power forecasting: Investigating the ability of deep learning models to capture low-level utility-scale photovoltaic system behaviour," *Applied Energy*, vol. 285, p. 116395, March 2021.

[17] W. Zhao, H. Zhang, J. Zheng, Y. Dai, L. Huang, W. Shang, and Y. Liang, "A point prediction method based automatic machine learning for day-ahead power output of multi-region photovoltaic plants," *Energy*, vol. 223, p. 120026, May 2021.

[18] Q. Li, X. Zhang, T. Ma, C. Jiao, H. Wang, and W. Hu, "A multi-step ahead photovoltaic power prediction model based on similar day, enhanced colliding bodies optimization, variational mode decomposition, and deep extreme learning machine," *Energy*, vol. 224, p. 120094, June 2021.

[19] F. Najibi, D. Apostolopoulou, and E. Alonso, "Enhanced performance Gaussian process regression for probabilistic short-term solar output forecast," *International Journal of Electrical Power & Energy Systems*, vol. 130, p. 106916, September 2021.

[20] T. E. Hoff and R. Perez, "Quantifying PV power output variability," *Solar Energy*, vol. 84, pp. 1782–1793, October 2010.

[21] ——, "Modeling PV fleet output variability," *Solar Energy*, vol. 86, pp. 2177–2189, August 2012.

[22] G. Wirth, E. Lorenz, A. Spring, G. Becker, R. Pardatscher, and R. Witzmann, "Modeling the maximum power output of a distributed PV fleet," *Progress in Photovoltaics: Research and Applications*, vol. 23, pp. 1164–1181, September 2015.

[23] J. Marcos, I. de la Parra, M. García, and L. Marroyo, "Simulating the variability of dispersed large PV plants," *Progress in Photovoltaics: Research and Applications*, vol. 24, pp. 680–691, May 2016.

[24] A. Sklar, "Fonctions de répartition à n dimensions et leurs marges," *Publications de l'Institut de Statistique de l'Université de Paris*, vol. 8, pp. 229–231, 1959.

[25] B. V. Vliet, "Abe Sklar's 'Fonctions de répartition à n dimensions et leurs marges': The original document and an English translation," *SSRN Electronic Journal*, 2023.

[26] T. Schmidt, "Coping with copulas," in *Copulas: From theory to application in finance*, J. Rank, Ed. University of California, 2007, pp. 3–34.

[27] A. Charpentier, J.-D. Fermanian, and O. Scaillet, "The estimation of copulas: Theory and practice," in *Copulas: From theory to application in finance*, J. Rank, Ed. University of California, 2007, pp. 35–61.

[28] A. J. Patton, "Copula-based models for financial time series," in *Handbook of Financial Time Series*. Springer, 2009, pp. 767–785.

[29] Y. Stander, D. Marais, and I. Botha, "Trading strategies with copulas," *Journal of Economic and Financial Sciences*, vol. 6, no. 1, pp. 83–107, 2013.

[30] P. Xu, D. Wang, V. P. Singh, Y. Wang, J. Wu, H. Lu, L. Wang, J. Liu, and J. Zhang, "Time-varying copula and design life level-based nonstationary risk analysis of extreme rainfall events," *Hydrology and Earth System Sciences Discussions*, pp. 1–59, 2019.

[31] Y. Zhao and M. Udell, "Missing value imputation for mixed data via Gaussian copula," in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2020, pp. 636–646.

[32] J. L. Schafer, "NORM users guide: Multiple imputation of incomplete multivariate data under a normal model," *University Park: The Methodology Center, Penn State*, 1999. [Online]. Available: https://scholarsphere.psu.edu/collections/v41687m23q

[33] Y. Zhao and M. Udell, "Matrix completion with quantified uncertainty through low rank Gaussian copula," *Adv. Neural Inf. Process. Syst.*, vol. 33, pp. 20 977–20 988, 2020.

[34] J. Munkhammar and J. Widén, "An autocorrelation-based copula model for producing realistic clear-sky index and photovoltaic power generation time-series," in *2017 IEEE 44th Photovoltaic Specialist Conference (PVSC)*. IEEE, 2017, pp. 3067–3072.

[35] W. Härdle and O. Linton, "Applied nonparametric methods," in *Handbook of Econometrics*, 4th ed., R. Engle and D. McFadden, Eds. Springer, 1994, pp. 767–785.

[36] A. Pagan and A. Ullah, *Nonparametric Econometrics*. Cambridge University Press, 1999.

[37] E. C. Brechmann and C. Czado, "COPAR-multivariate time series modeling using the copula autoregressive model," *Applied Stochastic Models in Business and Industry*, vol. 31, pp. 495–514, July 2015.

[38] S. Chen and R. Gopinath, "Gaussianization," *Advances in Neural Information Processing Systems*, vol. 13, 2000.

[39] C. W. Huang, R. T. Q. Chen, C. Tsirigotis, and A. Courville, "Convex potential flows: Universal probability distributions with optimal transport and convex optimization," *arXiv preprint*, 2020.

[40] H. Liao and J. He, "Jacobian determinant of normalizing flows," *arXiv preprint arXiv:2102.06539*, 2021.

[41] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge University Press, 2009.

[42] S. Diamond and S. Boyd, "CVXPY: A Python-embedded modeling language for convex optimization," *Journal of Machine Learning Research*, vol. 17, no. 83, pp. 1–5, 2016.

[43] A. Agrawal, R. Verschueren, S. Diamond, and S. Boyd, "A rewriting system for convex optimization problems," *Journal of Control and Decision*, vol. 5, no. 1, pp. 42–60, 2018.

[44] S. Boyd and L. Vandenberghe, *Introduction to Applied Linear Algebra*. Cambridge university press, 2018.

[45] J. Tuck and S. Boyd, "Fitting Laplacian regularized stratified Gaussian models," *SSRN Electronic Journal*, vol. 1, pp. 1–24, 2020.

[46] J. Tuck, S. Barratt, and S. Boyd, "A distributed method for fitting Laplacian regularized stratified models," *Journal of Machine Learning Research*, vol. 22, pp. 1–37, 2021.

[47] B. Meyers and S. Boyd, "Signal decomposition using masked proximal operators," *Foundations and Trends in Signal Processing*, vol. 17, no. 1, pp. 1–78, 2023.

[48] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, ser. Springer Series in Statistics. New York, NY: Springer New York, dec 2009. [Online]. Available: https://hastie.su.domains/ElemStatLearn/

[49] R. Koenker, *Quantile Regression*. Cambridge University Press, July 2005.

[50] ——, "Quantile regression: 40 years on," *Annual Review of Economics*, vol. 9, pp. 155–176, August 2017.