Stable Training of Probabilistic Models Using the Leave-One-Out Maximum Log-Likelihood Objective

Kutay Bölat, Simon H. Tindemans, Peter Palensky
Department of Electrical Sustainable Energy
Technische Universiteit Delft
Delft, The Netherlands
{K.Bolat, S.H.Tindemans, P.Palensky}@tudelft.nl

***Abstract*—Probabilistic modelling of power systems operation and planning processes depends on data-driven methods, which require sufficiently large datasets. When historical data lacks this, it is desired to model the underlying data generation mechanism as a probability distribution to assess the data quality and generate more data, if needed. Kernel density estimation (KDE) based models are popular choices for this task, but they fail to adapt to data regions with varying densities. In this paper, an *adaptive KDE* model is employed to circumvent this, where each kernel in the model has an individual bandwidth. The *leave-one-out maximum log-likelihood* (LOO-MLL) criterion is proposed to prevent the singular solutions that the regular MLL criterion gives rise to, and it is proven that LOO-MLL prevents these. Relying on this guaranteed robustness, the model is extended by adjustable weights for the kernels. In addition, a *modified expectation-maximization* algorithm is employed to accelerate the optimization speed reliably. The performance of the proposed method and models are exhibited on two power systems datasets using different statistical tests and by comparison with Gaussian mixture models. Results show that the proposed models have promising performance, in addition to their singularity prevention guarantees.**

***Index Terms*—adaptive kernel density estimation, expectation-maximization, leave-one-out, power systems data, probabilistic models**

I. Introduction

Today's power systems exhibit unprecedented levels of variability, especially due to the high penetration of renewable energy systems. Understanding these variabilities is crucial for the operation and planning of power systems because good models that represent these uncertainties can aid the decision-making process of system operators. Data-driven methods are the go-to approaches for such modelling tasks, but the effectiveness of these methods depends on the data quality, including its abundance, representativeness and health (missing values, outliers, etc.). Thus, assessing and improving the quality of real-life data is crucial for effective modelling.

One way to achieve these goals is the *data-driven probabilistic modelling* of the data, which aims to find the closest distribution to the unobservable real-life distribution that generated the data as a probability density function (*pdf*). Fig. 1 illustrates this process and Fig. 2 indicates both the complexity of dependencies and the ability to use such a model to generate more data. These can aid various data-driven applications such as security analysis [1], and anomaly detection [2].

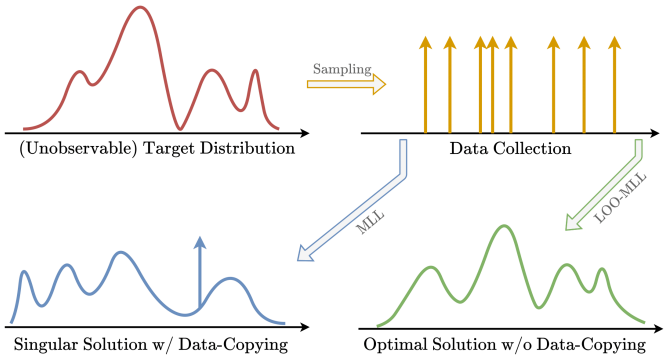


Fig. 1. Overall objective of probabilistic modelling. Objective functions (MLL, LOO-MLL) can only use the data points. MLL directly targets the dataset, and it copies a random data point (singularity). LOO-MLL avoids this, and the resulting model is similar to the real target distribution.

There is a wide variety of methods for the density estimation problem such as copulas [1], Gaussian mixture models (GMMs) [3], and variational autoencoders [4]. One of the most common among these is kernel density estimation (KDE) [3], which relies on placing kernels centred on data points and averaging them to form a pdf. However, the regular KDE model has an identical bandwidth parameter for all its kernels. This could lead to (1) noisy samples from the probabilistic model and (2) suboptimal estimation of the pdf in low-density regions where the data is scarce. This challenge can be overcome by assigning individual bandwidths for each data point with respect to their relative locations among each other [5], [6]. Thus, this *adaptive* KDE (A-KDE) model requires an optimization objective for its bandwidths to be assigned.

The maximum log-likelihood (MLL) criterion is one of the most well-established objectives for optimizing probabilistic models. Yet, the high flexibility of the adaptive KDE model leads to a phenomenon called *data-copying* [7]. This occurs when the A-KDE model is optimized to the extent that at

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 956433.

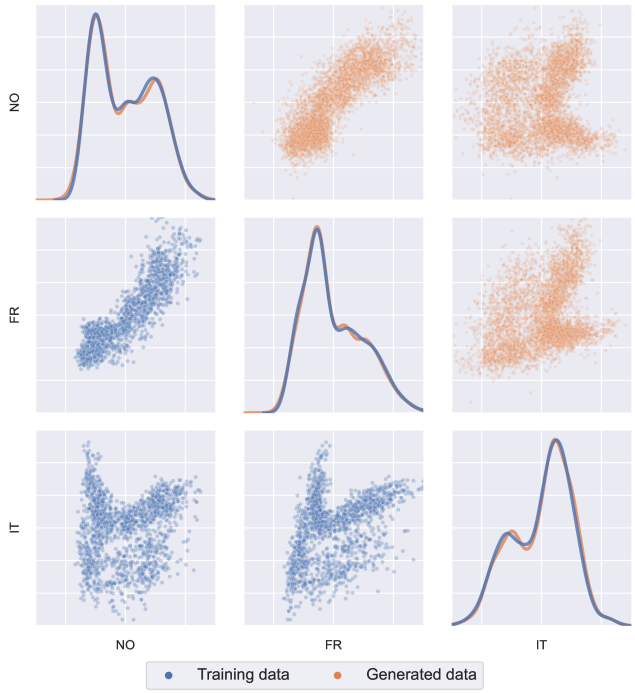


Fig. 2. Visual comparison between the training and the generated daily consumption data for three countries (NO, FR, IT) from the Europe dataset (Section IV-A), illustrating complex dependencies.

least one bandwidth converges to zero and re-produces the data point as seen in Fig. 1 (lower left) and results in a *singular* solution that causes the optimization algorithm to stop while the rest of distribution is arbitrarily shaped. Please note that most, if not all, of the highly flexible probabilistic models, such as GMMs and variational autoencoders, are prone to this phenomenon [7].

Avoiding data-copying can be crucial for applications where the probabilistic modelling is part of a pipeline. For instance, a reliability assessment pipeline can have its probabilistic asset modelling task automated for a smoother decision-making process, or edge computing devices in substations can utilize probabilistic models to automatically update their detection algorithms. These applications necessitate guarantees of non-singular solutions for providing reliable data processing pipelines, especially if the resources (computation power, time, battery, etc.) are limited.

Contributions. In this work, we explore a singularity mitigation strategy called the leave-one-out (LOO) MLL criterion using KDE-based models [6] in a similar fashion to the jackknife estimation method in statistics [8][1]. The contributions are as follows.

- We prove that KDE-based models converge to singular solutions when the regular MLL criterion is employed.
- We introduce the LOO-MLL objective to the KDE-based models and prove that it results in well-behaved and robust solutions.
- We propose π-KDE as a more flexible extension of A-KDE by integrating kernel weights into the model.
- We propose a modified expectation-maximization (EM) optimization procedure to exploit the advantages of EM.
- We developed a testing procedure to compare the performance of probabilistic models quantitatively.

[1]The jackknife estimation method systematically resamples the dataset by leaving out one observation at a time to reveal the effect of individual data points on the estimator. It can be used for cross-validation and bias-variance estimation.

II. Problem Definition

A. Motivating Examples

We start by giving some motivating examples of probabilistic modelling for power system applications.

1) Synthetic data generation: Smart meter data holds great potential for power systems operation and planning. Unfortunately, accessing these data is not generally possible due to privacy concerns. Moreover, even if the data is accessible, it might not be abundant enough due to its historic nature. These challenges can be overcome by data-driven modelling of the smart meter data at hand so that new datasets following a similar distribution can be generated by taking -effectively unlimited- samples from the probabilistic model. However, the data-copying phenomenon can hinder the privacy-preservation and expressiveness of the model due to the generation of exact replicas of certain data points in the original dataset.

2) Rare-event sampling: Certain events, such as extreme weather conditions, are crucial for various power systems applications like reliability assessment and predictive maintenance. Data points belonging to these events tend to be scarce, which may undermine the results of the application by creating a bias towards a few specific usual events. Probabilistic modelling can remedy this if the distribution's tail is modelled properly so one can take samples from the tail and enrich the rare-event data with unseen rare events. Yet, copying the rare-event data in the dataset interferes with the generalization of the model in the tail regions and inhibits the enrichment.

B. Preliminaries

1) Maximum Log-likelihood Criterion: Let $p(\mathbf{x};\theta) : \mathbb{R}^d \to \mathbb{R}_{\geq 0}$ represent the (multi-dimensional) pdf that we employ as our parametric distribution model where $\theta = \{\theta_a, \theta_f\}$ is the union of adjustable (θ_a) and fixed (θ_f) model parameters. Using the dataset $\mathcal{X} = \{\mathbf{x}_i\}_{i=1}^N$, we aim to find the best model that captures the underlying data generation process. The most common approach to represent this aim as a mathematical objective is using the MLL criterion,

$$\theta_a^* = \underset{\theta_a}{\operatorname{argmax}} \frac{1}{N} \sum_i \log p(\mathbf{x}_i;\theta), \tag{1}$$

where we hereon call the $\frac{1}{N}\sum_i \log p(\mathbf{x}_i;\theta)$ term as the *total log-likelihood*. Intuitively, this criteria incentivizes the model to cover as many data points as possible in its high-density regions since likelihood is an indicator of expected frequency.

2) Kernel Density Estimation: KDE is a methodology that utilizes the data points in the dataset to parameterize the probabilistic model. This is accomplished by placing *identical* kernels centred on data points and averaging them. For conciseness, we focus on Gaussian kernels in this work. Thus, the pdf of the KDE-based model can be written as

$$p_{\text{KDE}}(\mathbf{x};\theta) = \frac{1}{N}\sum_j \mathcal{N}(\mathbf{x};\mu_j = \mathbf{x}_j, \Sigma_j = \sigma^2\mathbf{I}) \tag{2}$$

which essentially is a mixture distribution model with uniform weights. Conventionally, the parameters of the KDE model are fixed, i.e. $\theta_a = \varnothing$, and the *bandwidth* parameter, $\sigma > 0$, is the same for all kernels. As a result, the applicability of KDE-based models to high-dimensional datasets is limited due to decreasing locality with increasing dimensionality [5], [6].

C. Adaptive Kernel Density Estimation

The aforementioned drawbacks of KDE models motivate us to employ *individual* bandwidths for each kernel to have the flexibility of adapting the coverage of the model locally. This adaptive KDE (A-KDE) model is defined as

$$p_{\text{A-KDE}}(\mathbf{x};\theta) = \frac{1}{N}\sum_j \mathcal{N}(\mathbf{x};\mu_j = \mathbf{x}_j, \Sigma_j = \sigma_j^2\mathbf{I}). \tag{3}$$

where $\sigma_j > 0,\ \forall j$. Additionally, we let the bandwidths be adjustable, i.e. $\theta_a = \{\sigma_j\}_{j=1}^N$.

D. Data-copying as a Singularity

The additional flexibility and adjustability that come with A-KDE encourage us to optimize θ_a according to the MLL criterion in (1). Unfortunately, the direct employment of this criterion as an objective function results in one or more bandwidth parameters converging to zero (*bandwidth-collapse*). Since a kernel with zero bandwidth contains no uncertainty and precisely represents the data point, we call this phenomenon *data-copying*.

Definition 1 (Data-copying)**.** *An A-KDE model copies a data point $\mathbf{x}_{j'} \in \mathcal{X}$ when $\sigma_{j'} \to 0^+$. Thus, the data-copying phenomenon occurs when $\exists j : \sigma_j \to 0^+$.*

Theorem 1. *An A-KDE model optimized by MLL objective copies at least one data point if and only if the total log-likelihood goes to infinity, i.e.*

$$\exists j : \sigma_j \to 0^+ \iff \sum_i \log p_{\text{A-KDE}}(\mathbf{x}_i;\theta) \to \infty.$$

The proof of the Theorem 1 can be found in App. A. This theorem implies that data-copying is a property of the global optimizer of the MLL objective, resulting in a *singular* solution for the density estimation problem.

Intuitively, the MLL objective drives the model to replicate the empirical data distribution by copying all the data points, i.e. $\sigma_j \to 0^+,\ \forall j$. Ideally, these bandwidth-collapses continue throughout the optimization until the full replication of the dataset. However, as we have shown, one bandwidth-collapse is enough to take the total log-likelihood to infinity. In practice, numerical optimization algorithms cannot handle infinite values and stop the optimization before the other bandwidths collapse. Thus, besides being singular, we can define the data-copying phenomenon also as an *unstable* solution.

III. Methodology

The aforementioned challenges that come with the increased flexibility of A-KDE models require a mitigation mechanism for healthy optimization. Since we want our method to be applicable to the problem without any prior assumptions regarding data, we rule out ad-hoc methods limiting the flexibility of the model, like regularization.

A. Leave-One-Out Maximum Log-Likelihood Objective for Adaptive Kernel Density Estimation

In order to mitigate the bandwidth-collapses, we should look at the root of the problem. Intuitively, it is more rewarding for the kernels to assign higher likelihoods to the data points that they centred on (*self-contribution*), which drives these kernels to ignore the surrounding data points. This can also be seen in the data-copying proof in App. A.

A natural solution to this problem would be to modify the MLL objective in a way that we leave these self-contributions out of the total log-likelihood, i.e.

$$\{\sigma_j^*\}_j = \underset{\{\sigma_j\}_j}{\text{argmax}} \sum_i \log \sum_{j\neq i} \mathcal{N}(\mathbf{x}_i;\mathbf{x}_j,\sigma_j^2\mathbf{I}). \tag{4}$$

We call this the LOO-MLL objective for A-KDE. We guarantee this objective solves the data-copying problem by the following theorem, and its proof can be found in App. B.

Theorem 2. *Data-copying cannot occur for any optimal solution for the modelling problem with A-KDE if the LOO-MLL objective is used and there are no repeating data points in the dataset.*

Note that the unique data points assumption holds almost surely (with probability one) for non-discrete data domains, so that the data-copying is not a problem for datasets drawn from continuous distributions. In addition, we also show that the instability problem that we encounter in the regular MLL objective does not occur when we employ the LOO-MLL objective.

Theorem 3. *The total log-likelihood in* (4) *is always bounded from above if there are no repeating data points in the dataset.*

Proof. Let us define $m := \min_{\{i,j:i\neq j\}}(\|\mathbf{x}_i - \mathbf{x}_j\|)$. Thanks to the no-replica data point assumption we have $m > 0$, and we can derive

$$\begin{aligned}\mathcal{N}(\mathbf{x}_i;\mathbf{x}_{j\neq i},\sigma_j^2\mathbf{I}) &\propto \sigma_j^{-d}\exp(-\frac{\|\mathbf{x}_i - \mathbf{x}_{j\neq i}\|^2}{2\sigma_j^2}) \\ &\leq \frac{1}{\sigma_j^d}\exp(\frac{-m^2}{2\sigma_j^2}) \leq \frac{d^{\frac{d}{2}}}{\exp(\frac{d}{2})}m^{-d} = c < \infty\end{aligned} \tag{5}$$

As a result,

$$\sum_i \log \sum_{j \neq i} \mathcal{N}(\mathbf{x}_i; \mathbf{x}_j, \sigma_j^2 \mathbf{I}) < N \log((N-1)c) < \infty \quad (6)$$

which concludes the proof. □

Consequently, employing LOO-MLL objective to non-discrete datasets almost surely guarantees the prevention of A-KDE's drawbacks, namely data-copying and instability.

B. π-Kernel Density Estimation

As we mentioned before, the KDE-based models are essentially mixture models, and A-KDE models are no exceptions. We can use this resemblance to extend their flexibility by introducing individual weights to each kernel as

$$p_{\pi\text{-KDE}}(\mathbf{x}; \theta) = \sum_j \pi_j \mathcal{N}(\mathbf{x}; \mathbf{x}_j, \sigma_j^2 \mathbf{I}) \quad (7)$$

where $\pi_j > 0$, $\sum_j \pi_j = 1$ and $\theta_a = \{\pi_j, \sigma_j\}_j$. Note that A-KDE is a special form of π-KDE with $\pi_j = \frac{1}{N}, \forall j$. Accordingly, the LOO-MLL objective for π-KDE models can be written in a similar manner too:

$$\{\pi_j^*, \sigma_j^*\}_j = \underset{\{\pi_j, \sigma_j\}_j}{\operatorname{argmax}} \sum_i \log \sum_{j \neq i} \pi_j \mathcal{N}(\mathbf{x}_i; \mathbf{x}_j, \sigma_j^2 \mathbf{I}). \quad (8)$$

This integration of the kernel weights introduces greater flexibility thanks to the higher number of parameters. Additionally, we hypothesize that the model's sensitivity to outliers is reduced by employing this method since, intuitively, the kernels belonging to these outlier data points tend to have smaller weights. The analysis of this claim is outside of the scope of this study, and we leave it for future work.

Corollary 3.1. *The theorems 1-3 apply to the π-KDE model and its related LOO-MLL objective.*

This corollary can be proved directly by including the weights to the derivations in the corresponding proofs. Intuitively, the π-KDE model employs a convex combination of the likelihoods in the A-KDE model and adapting this convexity to the given proofs does not change the results.

C. Modified Expectation-Maximization Algorithm

Until now, no specific optimization algorithm has been indicated to find the optimal solutions for the aforementioned LOO-MLL problems in III-A and III-B. Because of their continuous nature, a wide variety of off-the-shelf automatic differentiation-based optimizers, such as Adam [9] are applicable to our problem. These optimizers provide a seamless first-order gradient-based optimization for a given model and objective function.

On the other hand, the A-KDE/π-KDE models are special cases of isotropic Gaussian mixture models where the centres are predetermined and fixed, suggesting that we can employ the EM algorithm [3] for its desirable properties and intuitive implementation. Thus, the expectation and maximization steps of the algorithm can easily be applied to the conventional MLL objective. However, we must modify this EM algorithm according to our LOO-MLL objective to obtain well-behaved and stable solutions.

Accordingly, we propose to use the following modified EM algorithm to iteratively maximize the LOO-MLL objective:

- **E-step:**

$$r_{ij} = \begin{cases} \frac{\pi_j \mathcal{N}(\mathbf{x}_i; \mathbf{x}_j, \sigma_j^2 \mathbf{I})}{\sum_{j' \neq i} \pi_{j'} \mathcal{N}(\mathbf{x}_i; \mathbf{x}_{j'}, \sigma_{j'}^2 \mathbf{I})} & : i \neq j \\ 0 & : i = j \end{cases} \quad (9)$$

- **M-step:**

$$\sigma_j^2 = \frac{1}{d} \frac{\sum_i r_{ij} \|\mathbf{x}_i - \mathbf{x}_j\|^2}{\sum_i r_{ij}} \quad (10)$$

$$\pi_j = \frac{1}{N} \sum_i r_{ij} \quad (11)$$

The weights are fixed to $\pi_j = \frac{1}{N}, \forall j$ for A-KDE. Note that the M-step remains the same as the M-step of the regular EM algorithm, thanks to the assignment of zero responsibilities to the self-kernels, i.e. $r_{ii} = 0$. This assignment is a representation of the LOO mechanism in a way that the data points have no effect on the optimization of their self-kernels.

IV. Experimentation

In order to exhibit the probabilistic modelling capabilities of A-KDE and π-KDE models, we run a number of experiments. This section describes the datasets, the model comparison strategy and the experiment settings.[2]

A. Datasets

We employ the data from ENTSO-E Transparency Platform[3][4] as the basis of our datasets. We curated two datasets using this platform[5], as given below.

1) Europe dataset: We used the daily averaged power consumption in MW of 15 European countries[6] between 2015-2020 to build this dataset, which resulted in 2099 data points with 15 features. In other words, each data point corresponds to one day in the given five-year period, and each feature represents the aggregated daily consumption of the corresponding country. Fig. 2 (blue points) visualizes this dataset using three countries' data, i.e. three data features.

2) Denmark dataset: We used the hourly averaged load and generation (solar, onshore wind and offshore wind) numbers (in MW) from the two bidding zones in Denmark in 2019 to build this dataset, which resulted in 8784 data points with 8 features. Therefore, each data point corresponds to an hour of a given day in 2019.

Please note that both datasets are treated as collections of *i.i.d.* snapshots in time, not as time series. This treatment is relevant for various energy systems applications such as scenario testing for (cross-border) energy market studies and statistical modelling of load levels interconnection capacity planning.

[2]https://github.com/kabolat/leave-one-out_maximum-log-likelihood.
[3]https://transparency.entsoe.eu/
[4]https://data.open-power-system-data.org/time_series/
[5]The datasets can be found in the shared code repository.
[6]AT, BE, CH, DE, DK, ES, FI, FR, GB, IE, IT, NL, NO, PT, SE.

B. Two-Step Model Comparison Strategy

Here, we introduce our model comparison strategy to assess the performance of the distribution models.

1) Sample Comparison Tests: In order to test the hypothesis of whether two sets of samples are coming from the same distribution or not, two multi-dimensional two-sample statistical tests were used: maximum mean discrepancy (MMD) [10] and energy [11] tests[7]. These aim to test the hypothesis if the *model samples* are coming from the same distribution as the *test samples*. The test samples consist of the data points that we held out during the training of the models.

2) Model Comparison Tests: Two sample tests are designed to give smaller scores when the null hypothesis (samples are coming from the same distribution) is more likely. However, the test scores themselves are not easy to interpret numerically. In other words, the score alone cannot say if the best model amongst the candidate models is a good model or not. To overcome this, first, we obtained baseline scores by performing two-sample tests that compared the training data with the test data.

However, one drawback of this approach is that we *randomly* split the training and test samples before the optimization. Thus, the obtained baseline score is merely a sample of a complex random process and can be misleading. Since conducting all the optimization procedures for different data splits is infeasible, we propose to use the random subsets of the train, test and generated sample sets to capture this effect as done in [12]. This Monte Carlo approach results in *samples of sample comparison scores*, that are used to compare models.

Samples of test scores from the models can be compared to baseline score samples by using different statistics to calculate the difference between these sample distributions. For this, we use three two-sample tests as *model comparison tests*: Kolmogorov-Smirnov (KS) [13], Cramér–von Mises (CvM)[8] [14], and simple mean difference (ΔMean) tests. These result in *model comparison scores* with smaller values indicating better performance.

Algorithm 1 contains a pseudo-algorithm of the two-step model comparison procedure. Here, $\mathcal{M}$, $\mathcal{X}^{\text{train}}$, $\mathcal{X}^{\text{test}}$, N^{MC} and r represents the set of the compared models, training set, test set, the number of Monte Carlo runs and the subsampling ratio, respectively. The operator $\overset{\text{n}}{\sim}$ means sampling n data points without replacement. As a result, we have a collection of model comparison scores $\mathcal{S}^{\text{M}}_{\text{T}^{\text{s}},\text{T}^{\text{m}}}$ for all model, sample comparison test and model comparison test triples, e.g. $\mathcal{S}^{\pi-\text{KDE}}_{\text{MMD,CvM}}$.

C. Experiment Settings

1) Dataset settings: Both datasets were randomly split into train and test sets with a ratio of 4:1 and normalized using z-score normalization.

2) Optimizer settings: Adam was used for the gradient-based optimization. The convergence thresholds were set to 10^{-4} for all of the optimization algorithms.

[7]https://github.com/josipd/torch-two-sample

[8]scipy.stats (v1.8.1) is used for KS and CvM tests.

Algorithm 1 Two-Step Model Comparison Procedure

Require: $\mathcal{M}$, $\mathcal{X}^{\text{train}}$, $\mathcal{X}^{\text{test}}$, $N^{\text{MC}}, N^{\text{model}} \in \mathbb{N}$, $r \in (0,1)$

$\mathcal{T}^{\text{S}} = \{\text{MMD}, \text{Energy}\}$ ▷ Sample comparison tests
$n \leftarrow r \times |\mathcal{X}^{\text{test}}|$ ▷ # subsamples
for all $\text{T}^{\text{s}} \in \mathcal{T}^{\text{S}}$ **do**
  $\mathcal{S}^{\text{base}}_{\text{T}^{\text{s}}} \leftarrow \{\}$
  for all $i \in 1, 2, \ldots, N^{\text{MC}}$ **do**
    $\mathcal{D}^{\text{test}} \overset{\text{n}}{\sim} \mathcal{X}^{\text{test}}$; $\mathcal{D}^{\text{base}} \overset{\text{n}}{\sim} \mathcal{X}^{\text{train}}$ ▷ Subsampling
    $\mathcal{S}^{\text{base}}_{\text{T}^{\text{s}}} \leftarrow \mathcal{S}^{\text{base}}_{\text{T}^{\text{s}}} \cup \text{T}^{\text{s}}(\mathcal{D}^{\text{test}}, \mathcal{D}^{\text{base}})$
    ▷ Collecting the baseline sample scores for test T^{s}
  end for
end for

for all $\text{M} \in \mathcal{M}$ **do**
  $\mathcal{X}^{\text{model}} \overset{N^{\text{model}}}{\sim} \text{M}$ ▷ Taking samples from the model
  for all $\text{T}^{\text{s}} \in \mathcal{T}^{\text{S}}$ **do**
    $\mathcal{S}^{\text{M}}_{\text{T}^{\text{s}}} \leftarrow \{\}$
    for all $i \in 1, 2, \ldots, N^{\text{MC}}$ **do**
      $\mathcal{D}^{\text{test}} \overset{\text{n}}{\sim} \mathcal{X}^{\text{test}}$; $\mathcal{D}^{\text{model}} \overset{\text{n}}{\sim} \mathcal{X}^{\text{model}}$ ▷ Subsampling
      $\mathcal{S}^{\text{M}}_{\text{T}^{\text{s}}} \leftarrow \mathcal{S}^{\text{M}}_{\text{T}^{\text{s}}} \cup \text{T}^{\text{s}}(\mathcal{D}^{\text{test}}, \mathcal{D}^{\text{model}})$
        ▷ Collecting the sample scores of M for T^{s}
    end for
  end for
end for

$\mathcal{T}^{\text{M}} = \{\text{KS}, \text{CvM}, \Delta\text{Mean}\}$ ▷ Model comparison tests
for all $(\text{T}^{\text{s}}, \text{T}^{\text{m}}, \text{M}) \in \mathcal{T}^{\text{S}} \times \mathcal{T}^{\text{M}} \times \mathcal{M}$ **do**
  $\mathcal{S}^{\text{M}}_{\text{T}^{\text{s}},\text{T}^{\text{m}}} \leftarrow \text{T}^{\text{m}}(\mathcal{S}^{\text{base}}_{\text{T}^{\text{s}}}, \mathcal{S}^{\text{M}}_{\text{T}^{\text{s}}})$
    ▷ Assigning model scores for every T^{s} and M
end for

3) Initialization settings: The initial bandwidths were assigned as 0.1 both for A-KDE and π-KDE models [9] and initial weights were assigned as $\frac{1}{N}$ for the π-KDE model.

4) Benchmark model settings: Full covariance GMMs were used as the model of comparison. Their numbers of components were set in a way that the total numbers of parameters were as close as possible to the numbers of parameters of A-KDE and π-KDE. The GMM models are denoted as GMM_{A} and GMM_{π}, respectively. Thus, the set of models becomes $\mathcal{M} = \{\text{A-KDE}, \pi\text{-KDE}, \text{GMM}_{\text{A}}, \text{GMM}_{\pi}\}$.

5) Test settings: The number of samples was set to the size of the training set for all models, i.e. $N^{\text{model}} = |\mathcal{X}^{\text{train}}|$. We chose a subsampling rate of $r = 0.5$ and set the number of Monte Carlo runs (N^{MC}) to 2000 and 1000 for Europe and Denmark datasets, respectively.

V. Results and Discussion

A. Comparison of Training Speed

First, we compared the speed differences between two optimization algorithms (modified EM and Adam) by measuring

[9]A logarithmic search was conducted for the initial bandwidths, and no value performs the best on all scores. Since the resulting scores are relatively close to each other, 0.1 was chosen as the most representative initial bandwidth value.

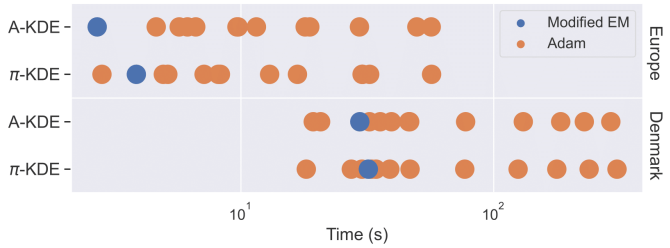


Fig. 3. Convergence times of the modified EM and Adam.

the elapsed times for convergence. Since the convergence speed of Adam also depends on the selection of the batch size and learning rate, we created a test grid that comprised combinations of several values of these hyperparameters[10].

The results are illustrated in Fig. 3. As can be seen, the modified EM algorithm is not necessarily faster than Adam. However, the fact that the abundance of hyperparameter combinations that result in slower convergences makes the modified EM algorithm more favourable thanks to the absence of hyperparameters. Moreover, please note that the results are given in the logarithmic scale. Thus, the convergence time of Adam is significantly longer than the modified EM for most combinations, while the difference is negligible when Adam is faster.

B. Estimation Performance Comparison

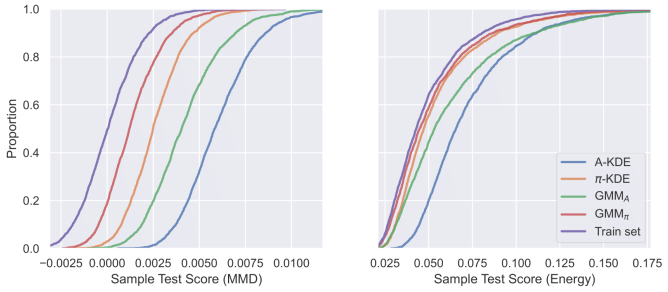
We trained all candidate models for each dataset. The empirical cumulative distribution functions (ECDFs) of the resulting sample test scores ($\mathcal{S}_{\mathrm{T^s}}^{\mathrm{M}}$ and $\mathcal{S}_{\mathrm{T^s}}^{\mathrm{base}}$) after the Monte Carlo runs are depicted in Fig. 4. The corresponding model comparison scores ($\mathcal{S}_{\mathrm{T^s},\mathrm{T^m}}^{\mathrm{M}}$) are tabulated in Table I. The final column indicates if the model is subject to singularity prevention.

In Fig. 4, we can see that the sample test scores of the models ($\mathcal{S}_{\mathrm{T^s}}^{\mathrm{M}}$) have similar variances with respect to their corresponding baseline sample test scores ($\mathcal{S}_{\mathrm{T^s}}^{\mathrm{base}}$). This eases the visual inspection by looking at the ordering of the ECDFs. Since we hypothesise that a lower score means better performance and that train and test data are drawn from the same distribution, having the baseline ECDFs on the leftmost for all orderings confirms our intuition.

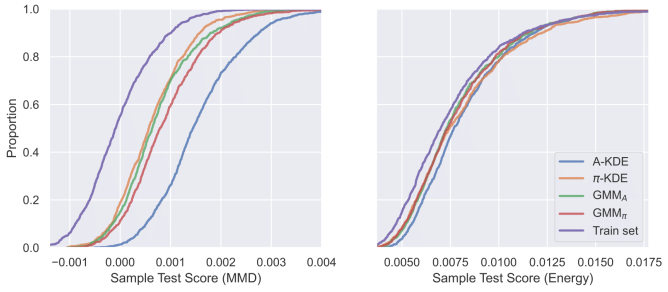
First, we see from Table I that A-KDE is consistently inferior to the π-KDE model. This supports the motivation of introducing kernel weights described in Section III-B.

The similarity of distributions in Fig. 4 suggests that all tested models are able to adequately represent the Denmark dataset. We interpret this as a result of the lower dimensionality and larger dataset size with respect to the Europe dataset. Nonetheless, π-KDE is more desirable in practice thanks to its singularity prevention guarantees. GMMs lack this prevention, and it is likely to encounter singularities, which we experienced occasionally during our experimentation.

[10]Batch size: [128, 256, 512, 1024]. Learning rate: [0.01, 0.05, 0.10]



(a)



(b)

Fig. 4. ECDFs of sample test scores for (a) Europe and (b) Denmark datasets.

TABLE I
MODEL COMPARISON SCORES ($\mathcal{S}_{\mathrm{T^s},\mathrm{T^M}}^{\mathrm{M}}$)

Dataset	Sample Comparison Test (T^s)	Model	Model Comparison Test (T^m) KS $\times 10^{-1}$	CvM	ΔMean $\times 10^{-3}$	Singularity Prevention
Europe	MMD	A-KDE	9.16	317.88	5.91	**Yes**
		π-KDE	5.82	167.79	2.50	**Yes**
		$\mathrm{GMM_A}$	7.63	256.11	4.10	No
		GMM_π	**3.08**	**52.45**	**1.25**	No
	Energy	A-KDE	4.46	95.96	23.99	**Yes**
		π-KDE	1.24	6.95	5.97	**Yes**
		$\mathrm{GMM_A}$	2.02	21.96	14.30	No
		GMM_π	**0.56**	**1.27**	**3.34**	No
Denmark	MMD	A-KDE	6.84	111.03	1.58	**Yes**
		π-KDE	**3.79**	**36.23**	**0.65**	**Yes**
		$\mathrm{GMM_A}$	4.19	42.52	0.75	No
		GMM_π	4.74	55.41	0.91	No
	Energy	A-KDE	1.57	5.51	0.68	**Yes**
		π-KDE	0.95	2.24	0.57	**Yes**
		$\mathrm{GMM_A}$	1.06	1.77	**0.34**	No
		GMM_π	**0.93**	**1.67**	**0.34**	No

For the Europe dataset, the best scores are obtained by the GMM_π model. However, the π-KDE model also shows acceptable performance for this dataset. Qualitatively, we can see from Fig. 2 that the complex nature of the marginals and '2-way marginals' of the selected countries are captured by the generated data. This makes the π-KDE model more favourable in cases where singularity prevention is crucial, like in edge-computing, in which the computation power is limited and re-running a failed GMM optimization might be costly in terms of time and energy. Similarly, (near) real-time applications might also require this singularity prevention due to the limited time budget to re-run a failed optimization attempt.

VI. Conclusion and Future Work

Probabilistic modelling of power systems data is crucial for the future of systems operation and planning. This work introduced the data-copying phenomenon, which burdens such modelling by causing singular solutions. KDE-based models are employed to investigate this effect in a mathematically rigorous way. LOO-MLL criterion is proposed as a solution, and singularity prevention is guaranteed for two KDE-based models. Moreover, a modified EM optimization procedure is proposed for reliable training of the models. The models, along with benchmark GMM models, are tested on two power systems datasets using a novel testing procedure. The results show that the proposed models have an adequate modelling performance besides having singularity prevention guarantees.

Even though the KDE-based models are convenient for mathematical analysis, their applicability is limited since the number of kernels can easily be overwhelming for large datasets. A pruning mechanism might be the solution for this. Also, the isotropic nature of kernels can result in noisy samples if the data lies in a lower dimensional manifold. We plan to extend this work to kernels with full-covariance matrices. Lastly, as mentioned before, singularity caused by data-copying is a common problem in more advanced models too [7], such as GMMs and variational autoencoders, so it is appealing to use the LOO-MLL in these models in future.

References

[1] I. Konstantelos, M. Sun, S. H. Tindemans, S. Issad, P. Panciatici, and G. Strbac, "Using vine copulas to generate representative system states for machine learning," *IEEE Transactions on Power Systems*, vol. 34, no. 1, pp. 225–235, 2018.

[2] C. Wang, S. H. Tindemans, and P. Palensky, "Improved anomaly detection and localization using whitening-enhanced autoencoders," *IEEE Transactions on Industrial Informatics*, 2023.

[3] K. P. Murphy, *Machine learning: a probabilistic perspective*. MIT press, 2012.

[4] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[5] S. R. Sain, *Adaptive kernel density estimation*. Rice University, 1994.

[6] C. van der Walt and E. Barnard, "Variable kernel density estimation in high-dimensional feature spaces," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 31, no. 1, 2017.

[7] C. Meehan, K. Chaudhuri, and S. Dasgupta, "A non-parametric test to detect data-copying in generative models," in *International Conference on Artificial Intelligence and Statistics*, 2020.

[8] A. McIntosh, "The jackknife estimation method," *arXiv preprint arXiv:1606.00497*, 2016.

[9] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[10] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola, "A kernel two-sample test," *The Journal of Machine Learning Research*, vol. 13, no. 1, pp. 723–773, 2012.

[11] G. J. Székely and M. L. Rizzo, "Energy statistics: A class of statistics based on distances," *Journal of statistical planning and inference*, vol. 143, no. 8, pp. 1249–1272, 2013.

[12] C. Wang, E. Sharifnia, Z. Gao, S. H. Tindemans, and P. Palensky, "Generating multivariate load states using a conditional variational autoencoder," *Electric Power Systems Research*, vol. 213, p. 108603, 2022.

[13] F. J. Massey Jr, "The kolmogorov-smirnov test for goodness of fit," *Journal of the American statistical Association*, vol. 46, no. 253, pp. 68–78, 1951.

[14] T. W. Anderson, "On the distribution of the two-sample cramer-von mises criterion," *The Annals of Mathematical Statistics*, pp. 1148–1159, 1962.

Appendix A
Proof of Data-copying

Proof. ($\Rightarrow$) Let $\sigma_{j'} \to 0^+$ while the rest of the bandwidths, $\{\sigma_j\}_{j\neq j'}$ have non-zero values. Plugging these into the scaled total log-likelihood without taking the limit results in

$$\begin{aligned}\sum_i \log \sum_j \mathcal{N}(\mathbf{x}_i;\mathbf{x}_j,\sigma_j^2\mathbf{I}) &\stackrel{c}{=} \sum_i \log\left[\frac{1}{\sigma_{j'}^d}\exp\left(\frac{-\Delta_{ij'}^2}{2\sigma_{j'}^2}\right)+c_i\right]\\ &= \log\left[\frac{1}{\sigma_{j'}^d}+c_{j'}\right]+\sum_{i\neq j'}\log\left[\left(\frac{1}{\sigma_{j'}^2}\right)^{\frac{d}{2}}\exp\left(\frac{-\Delta_{ij'}^2}{2\sigma_{j'}^2}\right)+c_i\right]\end{aligned} \tag{12}$$

where $c_i \coloneqq \sum_{j\neq j'}\mathcal{N}(\mathbf{x}_i;\mathbf{x}_j,\sigma_j^2\mathbf{I})$, $\Delta_{ij} \coloneqq \|\mathbf{x}_i-\mathbf{x}_j\|$ and $\stackrel{c}{=}$ means equal up to a constant. Taking the limit $\sigma_{j'}\to 0^+$, we get

$$\begin{aligned}&\lim_{\sigma_{j'}\to 0^+}\sum_i \log\sum_j \mathcal{N}(\mathbf{x}_i;\mathbf{x}_j,\sigma_j^2\mathbf{I}) =\\ &-d\lim_{\sigma_{j'}\to 0^+}\log(\sigma_{j'})+\sum_{i\neq j'}\log(0+c_i)\to\infty\end{aligned} \tag{13}$$

assuming there are no replica points in the dataset, i.e. $\Delta_{ij'}\neq 0,\forall i$. Otherwise, the second term also goes to infinity. In both cases, the total limit diverges to infinity, concluding the proof of the only if part.

($\Leftarrow$) Let us assume the opposite of the conclusion, i.e. $\exists\varepsilon>0:\sigma_j>\varepsilon,\forall j$. This makes the likelihoods of the datapoints under every kernel finite, i.e. $\exists c\in\mathbb{R}^+:\mathcal{N}(\mathbf{x}_i|\mathbf{x}_j,\sigma_j^2\mathbf{I})\leq c,\forall i,j$. This also results in an upper-bounded total log-likelihood

$$\frac{1}{N}\sum_i\log\sum_j\mathcal{N}(\mathbf{x}_i;\mathbf{x}_j,\sigma_j^2\mathbf{I})\leq\log Nc \tag{14}$$

which contradicts the initial statement and concludes the proof. □

Appendix B
Proof of Data-copying Prevention by LOO-MLL

Proof. The assumption that there are no repeating data points implies $\min_{\{i,j:i\neq j\}}(\|\mathbf{x}_i-\mathbf{x}_j\|)>0$. The remainder of the proposition can be formulated as

$$\begin{aligned}\{\sigma_j^*\}_j &= \operatorname*{argmax}_{\{\sigma_j\}_j}\sum_i\log\sum_{j\neq i}\mathcal{N}(\mathbf{x}_i;\mathbf{x}_j,\sigma_j^2\mathbf{I})\\ &\implies\exists\varepsilon>0:\sigma_j^*\geq\varepsilon,\forall j\end{aligned} \tag{15}$$

and express the negation as

$$\begin{aligned}&\forall\varepsilon>0:\exists j:\sigma_j^*<\varepsilon\ \wedge\\ &g_j|_{\{\sigma_k^*\}_k} \coloneqq \left.\frac{\partial\sum_i\log\sum_{k\neq i}\exp(f_{ik})}{\partial\sigma_j}\right|_{\{\sigma_k^*\}_k}\leq 0,\end{aligned} \tag{16}$$

where $f_{ik}=-d\log\sigma_k-\frac{\Delta_{ik}^2}{2\sigma_k^2}$ and $\Delta_{ik}=\|\mathbf{x}_i-\mathbf{x}_k\|$. If we prove that (16) results in a contradiction, it concludes the proof. Note that the inequality in the (local) optimality condition covers the candidate solutions on the boundaries.

The gradient expression g_j can be derived as

$$g_j = \sum_{i \neq j} \left(\frac{\Delta_{ij}^2}{\sigma_j^3} - \frac{d}{\sigma_j} \right) w_{ij} \tag{17}$$

where $w_{ij} = \frac{\exp(f_{ij})}{\sum_{k \neq i} \exp(f_{ik})} \in (0,1)$. Let us assume that $\sigma_{j'}^* < \varepsilon$ and express $g_{j'}^* = g_{j'}|_{\{\sigma_k^*\}_k} \leq 0$ using (17) as

$$\begin{aligned} \frac{1}{\varepsilon^2} < \frac{1}{\sigma_{j'}^{*2}} \leq \frac{d \sum_{i \neq j'} w_{ij'}^*}{\sum_{i \neq j'} \Delta_{ij'}^2 w_{ij'}^*} = \frac{d}{\sum_{i \neq j'} \frac{\Delta_{ij'}^2}{1 + \sum_{l \neq i, j'} w_{lj'}}} \\ < \frac{d(N-1)}{\sum_{i \neq j'} \Delta_{ij'}^2} < \frac{d(N-1)}{N \min(\{\Delta_{ij'}^2\}_{i \neq j'})}. \end{aligned} \tag{18}$$

Thus, the optimality statement in (16) takes the form of

$$\forall \varepsilon > 0 : \exists j : \frac{N}{d(N-1)} \min(\{\Delta_{ij}^2\}_{i \neq j}) < \varepsilon^2. \tag{19}$$

This is a contradiction as long as the non-repeating data point assumption ($\min_{\{i,j:i \neq j\}}(\|\mathbf{x}_i - \mathbf{x}_j\|) > 0$) holds.

□