Supplementary Material for "Hidden Markov Models for Wind Farm Power Output"

Debarati Bhaumik, Daan Crommelin, Stella Kapodistria, and Bert Zwart

I. Scatterplots of wind turbine power output

The power output between the various wind turbines located in the same wind park (or in geographic proximity) is highly dependent. As evidence to this dependency structure, we plot the scatterplots of the power outputs of two wind turbines. In Fig. 1, we present the scatterplots of the power output of wind turbine 6 and those of wind turbines 1-5, respectively.

II. General description of HMM

A hidden Markov model can be viewed as a tool for representing probability distributions over sequences of observations. Let Y_t denote the observation at time t, $t = 1, 2, \ldots$ (we assume here the discrete time case), with state-space $\mathcal{N}$. The HMM gets its name from two defining properties: First, it is assumed that the observation at time t was generated by some hidden process X_t, $t = 1, 2, \ldots$, with state-space $\mathcal{M}$. Second, it is assumed that the state of the hidden process satisfies the Markov property. More concretely, under the standard HMM formalism, for a given set of observations $(y_1, \ldots, y_T)$, the stochastic processes $\{X_t,\, t = 1, 2, \ldots\}$ and $\{Y_t,\, t = 1, 2, \ldots\}$ are governed by the one step transition probability matrix $\boldsymbol{A}$ and the emission distribution $\boldsymbol{B}$ (also known as output probabilities):

$$\begin{cases} A_{i,j} = \mathbb{P}(X_{t+1} = j | X_t = i), \\ B_i(y_t) = \mathbb{P}(Y_t = y_t | X_t = i), \end{cases} \tag{1}$$

$\forall i, j \in \mathcal{M}$. For $t = 1$, the initial distribution of $\{X_t,\, t = 1, 2, \ldots\}$ is given by $\pi_i = \mathbb{P}(X_1 = i)$. A schematic diagram showing the dependence of the random variables $\{(X_t, Y_t),\, t = 1, 2 \ldots\}$ of the HMM described by (1) is presented in Fig. 2. However, the structure of the emission distribution $\boldsymbol{B}$ varies depending on the problem under consideration. Note that HMMs can also be viewed as a particular kind of linear neural networks, [5, page 312].

III. Expectation-Maximization (EM) algorithm

The EM algorithm is a general method for finding the maximum-likelihood parameter estimates of an underlying distribution given a data set which has incomplete values (see *[22], [24], [26], [28] references of main text*). Given the sequence of observations for all the wind turbine power outputs, $\mathbf{y} = [\mathbf{y}^1, \ldots, \mathbf{y}^W]$ where $\mathbf{y}^w = [y_1^w, \ldots, y_T^w] = [\tilde{P}_1^w, \ldots, \tilde{P}_T^w]$, our goal is to estimate the parameters of the model, λ. In order to do so we will maximize the joint likelihood function for the model given the complete data set of the output processes and the hidden process.

A. Likelihood Function

For a given model λ, the joint likelihood function for the model given the complete data set is the joint probability distribution for the observation and the hidden state sequences

$$\begin{aligned} \mathcal{L}(\lambda|\mathbf{y}, \mathbf{x}) &= \mathbb{P}(\mathbf{Y} = \mathbf{y}, \mathbf{X} = \mathbf{x}|\lambda) \\ &= \pi_{x_1} \prod_{w=1}^{W} \rho_{y_1^w}^{w,x_1} \prod_{t=2}^{T} \big[A_{x_{t-1}x_t} \prod_{w=1}^{W} L_{y_{t-1}y_t}^{w,x_t}\big]. \end{aligned} \tag{2}$$

The above equation can be factorized as a result of the Markov property, cf. *(3) of the main text*.

The EM algorithm for parameter estimation of HMM given a set of observations is known as the Baum-Welch (BW) algorithm (*[27] in the reference of main text*). The EM algorithm first finds the expected value of the complete data set log-likelihood with respect to the hidden data set $\mathbf{X} = \mathbf{x}$ given the observed data $\mathbf{Y} = \mathbf{y}$ and the current parameter estimates, in the *expectation-step*, i.e.,

$$\begin{aligned} &\mathcal{Q}(\lambda, \lambda_k) = \mathbf{E}\big[\mathbb{P}(\mathbf{Y} = \mathbf{y}, \mathbf{X} = \mathbf{x}|\lambda) | \mathbf{Y} = \mathbf{y}, \lambda_k\big] \\ &= \sum_{\mathbf{x} \in \mathcal{M}^T} \log(\mathbb{P}(\mathbf{Y} = \mathbf{y}, \mathbf{X} = \mathbf{x}|\lambda)) \mathbb{P}(\mathbf{Y} = \mathbf{y}, \mathbf{X} = \mathbf{x}|\lambda_k), \end{aligned} \tag{3}$$

where λ_k is the current set of parameters estimates used to calculate the expectation $\mathcal{Q}$ and λ is a new set of parameters. A key element of the EM algorithm is to optimize λ in order to increase $\mathcal{Q}$. A detailed discussion of expression (3) is given in reference *[28] in the main text*.

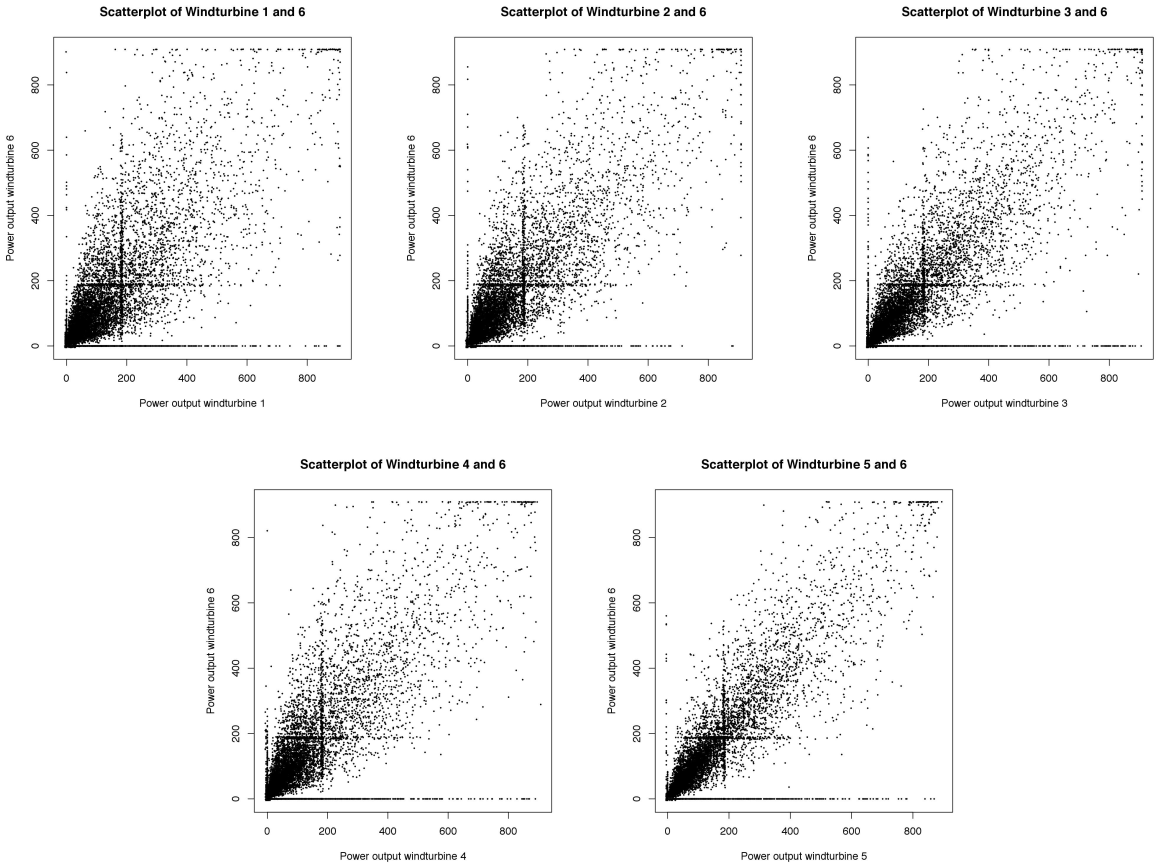

Fig. 1: Scatter plots of power output of wind turbine 6 with wind turbines 1-5

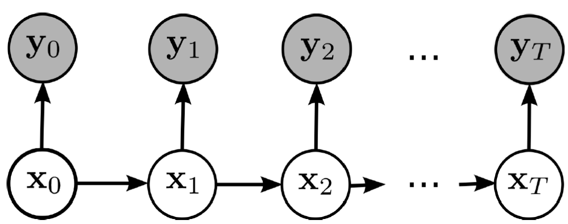

Fig. 2: Schematic diagram of the HMM described in (1)

The *maximization-step* determines the next iterate λ_{k+1} by maximizing the expectation Q, i.e.

$$\lambda_{k+1} = \underset{\lambda}{\text{argmax}}\ \mathcal{Q}(\lambda, \lambda_k). \tag{4}$$

The maximization step guarantees that $\mathcal{L}(\lambda_{k+1}|\mathbf{y}, \mathbf{x}) \geq \mathcal{L}(\lambda_k|\mathbf{y}, \mathbf{x})$. The expectation and maximization steps are repeated until the desired convergence is reached. For literature on the convergence of the EM algorithm, see *[29], [30] of main text*.

B. Parameter estimation

In this section we present expressions for the parameter estimates λ given the observed data $\mathbf{y}$ and the hidden sequence $\mathbf{x} = [x_1, \ldots, x_T]$. For convenience we will denote λ_k, the old parameter set, as λ' and the next iterate of the parameter set λ_{k+1} as λ. First we find the expectation function using (2), the expectation function (3) can be expanded as

$$\mathcal{Q}(\lambda, \lambda') = \sum_{\mathbf{x} \in \mathcal{M}^T} \big(\log \pi_{x_1}\big)\mathbb{P}(\mathbf{Y} = \mathbf{y}, \mathbf{X} = \mathbf{x}|\lambda') + \sum_{w=1}^{W} \Big[\sum_{\mathbf{x} \in \mathcal{M}^T} \big(\log \rho_{y_1^w}^{w,x_1}\big)\mathbb{P}(\mathbf{Y} = \mathbf{y}, \mathbf{X} = \mathbf{x}|\lambda')\Big]$$

$$+\sum_{\mathbf{x}\in\mathcal{M}^T}\big(\sum_{t=2}^{T}\log A_{x_{t-1}x_t}\big)\mathbb{P}(\mathbf{Y}=\mathbf{y},\mathbf{X}=\mathbf{x}|\lambda')$$

$$+\sum_{w=1}^{W}\Big[\sum_{\mathbf{x}\in\mathcal{M}^T}\big(\sum_{t=2}^{T}\log L^{w,x_t}_{y^w_{t-1}y^w_t}\big)\mathbb{P}(\mathbf{Y}=\mathbf{y},\mathbf{X}=\mathbf{x}|\lambda')\Big]. \tag{5}$$

Since the parameters we wish to optimize can be independently factorized into explicit terms as shown in the equation above, we can optimize each term individually using Lagrange multipliers.

a) First term, π: The first term of (5) can be written as

$$\sum_{\mathbf{x}\in\mathcal{M}^T}\big(\log\pi_{x_1}\big)\mathbb{P}(\mathbf{Y}=\mathbf{y},\mathbf{X}=\mathbf{x}|\lambda')$$
$$=\sum_{i=1}^{M}\sum_{(x_2,\dots,x_T)\in\mathcal{M}^{T-1}}(\log\pi_i)\cdot\mathbb{P}(\mathbf{Y}=\mathbf{y},$$
$$X_1=i,X_2=x_2,\dots X_T=x_T|\lambda')$$
$$=\sum_{i=1}^{M}(\log\pi_i)\mathbb{P}(\mathbf{Y}=\mathbf{y},X_1=i|\lambda').$$

Using the Lagrange multiplier γ and the constraint that $\sum_{i=1}^{M}\pi_i=1$, we set the derivative equal to zero, i.e.,

$$\frac{\partial}{\partial\pi_i}\Big(\sum_{i=1}^{M}(\log\pi_i)\mathbb{P}(\mathbf{Y}=\mathbf{y},X_1=i|\lambda')-\gamma\big(\sum_{i=1}^{M}\pi_i-1\big)\Big)=0.$$

Taking the derivative and using $\sum_{i=1}^{M}\pi_i=1$ we get $\gamma=\mathbb{P}(\mathbf{Y}=\mathbf{y}|\lambda')$ and for π_i we get,

$$\pi_i=\frac{\mathbb{P}(\mathbf{Y}=\mathbf{y},X_1=i|\lambda')}{\mathbb{P}(\mathbf{Y}=\mathbf{y}|\lambda')}. \tag{6}$$

b) Second term, $\rho^{w,i}$: The second term of (5) has a sum in w. We solve for the the w^{th} term inside the sum,

$$\sum_{\mathbf{x}\in\mathcal{M}^T}\big(\log\rho^{w,x_1}_{y_1{}^w}\big)\mathbb{P}(\mathbf{Y}=\mathbf{y},\mathbf{X}=\mathbf{x}|\lambda')$$
$$=\sum_{i=1}^{M}(\log\rho^{w,i}_{y_1^w})\mathbb{P}(\mathbf{Y}=\mathbf{y},X_1=i|\lambda') \quad .$$

We have M constraint equations $\sum_{\mu=1}^{N}\rho^{w,i}_{\mu}=1$, as $i\in\mathcal{M}$. Hence we have M Lagrange multipliers. Setting the derivative to zero and using $\sum_{\mu=1}^{N}\rho^{w,i}_{\mu}=1$, we get

$$\rho^{w,i}_{\mu}=\frac{\mathbb{1}(Y_1^w=\mu)\mathbb{P}(\mathbf{Y}=\mathbf{y},X_1=i|\lambda')}{\mathbb{P}(\mathbf{Y}=\mathbf{y},X_1=i|\lambda')}=\mathbb{1}(Y_1^w=\mu). \tag{7}$$

c) Third term, A: The third term of (5) can be written as

$$\sum_{\mathbf{x}\in\mathcal{M}^T}\big(\sum_{t=2}^{T}\log A_{x_{t-1}x_t}\big)\mathbb{P}(\mathbf{Y}=\mathbf{y},\mathbf{X}=\mathbf{x}|\lambda')=$$
$$\sum_{t=2}^{T}\sum_{i=1}^{M}\sum_{j=1}^{M}(\log A_{ij})\mathbb{P}(\mathbf{Y}=\mathbf{y},X_{t-1}=i,X_t=j|\lambda'). \tag{8}$$

We have M constraint equations $\sum_{k=1}^{M}a_{lk}=1$, as $l\in\mathcal{M}$. Hence we need M Lagrange multipliers. Setting the derivative to zero and using $\sum_{k=1}^{M}a_{lk}=1$, we get

$$A_{ij}=\frac{\sum_{t=2}^{T}\mathbb{P}(\mathbf{Y}=\mathbf{y},X_{t-1}=i,X_t=j|\lambda')}{\sum_{t=2}^{T}\mathbb{P}(\mathbf{Y}=\mathbf{y},X_{t-1}=i|\lambda')}. \tag{9}$$

d) Fourth term, $L^{w,i}$: We solve for the w^{th} term inside the sum of the fourth term of (5),

$$\sum_{\mathbf{x}\in\mathcal{M}^T}\big(\sum_{t=2}^{T}\log L^{w,x_t}_{y^w_{t-1}y^w_t}\big)\mathbb{P}(\mathbf{Y}=\mathbf{y},\mathbf{X}=\mathbf{x}|\lambda')$$
$$=\sum_{t=2}^{T}\sum_{i=1}^{M}(\log L^{w,i}_{y^w_{t-1}y^w_t})\mathbb{P}(\mathbf{Y}=\mathbf{y},X_t=i|\lambda'). \tag{10}$$

Solving for $L^{w,i}_{\mu\nu}$ using the Lagrange multipliers we get,

$$L^{w,i}_{\mu\nu}=\frac{\sum_{t=2}^{T}\mathbb{1}(Y^w_{t-1}=\mu)\mathbb{1}(Y^w_t=\nu)\mathbb{P}(\mathbf{Y}=\mathbf{y},X_t=i|\lambda')}{\sum_{t=2}^{T}\mathbb{1}(Y^w_{t-1}=\mu)\mathbb{P}(\mathbf{Y}=\mathbf{y},X_t=i|\lambda')}. \tag{11}$$

Hence, the parameter estimates are given by

$$\begin{cases}\widehat{\pi}_i=\frac{\mathbb{P}(\mathbf{Y}=\mathbf{y},X_1=i|\lambda')}{\mathbb{P}(\mathbf{Y}=\mathbf{y}|\lambda')},\\ \widehat{A}_{ij}=\frac{\sum_{t=2}^{T}\mathbb{P}(\mathbf{Y}=\mathbf{y},X_{t-1}=i,X_t=j|\lambda')}{\sum_{t=2}^{T}\mathbb{P}(\mathbf{Y}=\mathbf{y},X_{t-1}=i|\lambda')},\\ \widehat{\rho}^{w,i}_{\mu}=\mathbb{1}(Y_1^w=\mu),\\ \widehat{L}^{w,i}_{\mu\nu}=\frac{\sum_{t=2}^{T}\mathbb{1}(Y^w_{t-1}=\mu)\mathbb{1}(Y_t^w=\nu)\mathbb{P}(\mathbf{Y}=\mathbf{y},X_t=i|\lambda')}{\sum_{t=2}^{T}\mathbb{1}(Y^w_{t-1}=\mu)\mathbb{P}(\mathbf{Y}=\mathbf{y},X_t=i|\lambda')}.\end{cases} \tag{12}$$

1) Forward-backward variables: For calculating the estimates in (12) it is convenient to introduce the so-called forward backward variables $\alpha_i(t),\beta_i(t),\ \forall i\in\mathcal{M}$ ([24], [30]). For lack of space we have dropped the random variable Y terms in front of the y terms,

$$\begin{cases}\alpha_i(t) = \mathbb{P}(y_1^1,\dots,y_t^1,\dots,y_1^W,\dots,y_t^W, X_t = i|\lambda'),\\ \beta_i(t) = \mathbb{P}(y_{t+1}^1,\dots,y_T^1,\dots,y_{t+1}^W,\dots,y_T^W|X_t = i,\\ \qquad y_t^1,\dots,y_t^w,\lambda').\end{cases} \tag{13}$$

These variables are computed recursively and the numerical effort grows linearly in T.

a) Forward variable recursion: For simplicity we drop the λ' term for now. From (13) we have for $t = 1$,

$$\begin{aligned}\alpha_i(1) &= \mathbb{P}(y_1^1,\dots,y_1^W, X_1 = i)\\ &= \mathbb{P}(y_1^1|y_1^2,\dots,y_1^W, X_1 = i)\cdot\mathbb{P}(y_1^2,\dots,y_1^W, X_1 = i)\\ &= \rho^{1,i}_{y_1^1}\cdots\rho^{W,i}_{y_1^W}\cdot\pi_i\end{aligned}$$

For deriving the recursion relation, we have

$$\begin{aligned}&\alpha_i(t) = \mathbb{P}(y_1^1,\dots,y_t^1,\dots,y_1^W,\dots,y_t^W, X_t = i)\\ &= \sum_{j=1}^M \mathbb{P}(y_1^1,\dots,y_t^1,\dots,y_1^W,\dots,y_t^W, X_{t-1} = j, X_t = i)\\ &= \sum_{j=1}^M \mathbb{P}(y_t^1|y_1^1,\dots,y_{t-1}^1,\dots,y_1^W,\dots,y_t^W, X_{t-1} = j,\\ &\qquad X_t = i)\cdot\mathbb{P}(y_1^1,\dots,y_{t-1}^1,\dots,y_1^W,\dots,y_t^W,\\ &\qquad X_{t-1} = j, X_t = i)\\ &= L^{1,i}_{y_{t-1}^1 y_t^1}\cdots L^{W,i}_{y_{t-1}^W y_t^W}\cdot\mathbb{P}(y_1^1,\dots,y_{t-1}^1,\dots,y_1^W,\\ &\qquad \dots,y_{t-1}^W, X_{t-1} = j)\cdot A_{ji}\\ &= \sum_{j=1}^M \alpha_j(t-1)A_{ji}[\prod_{w=1}^W L^{w,i}_{y_{t-1}^w y_t^w}].\end{aligned} \tag{14}$$

b) Backward equation: From (13) we have, $\beta_i(T) = 1$. We have,

$$\begin{aligned}&\beta_i(t) = \mathbb{P}(y_{t+1}^1,\dots,y_T^1,\dots,y_{t+1}^W,\dots,y_T^W|X_t = i,\\ &\qquad y_t^1,\dots,y_t^W)\\ &= \sum_{\{x_{t+1},\dots,x_T\}\in\mathcal{M}^{T-t}} \mathbb{P}(y_{t+1}^1,\dots,y_T^1,\dots,y_{t+1}^W,\dots,y_T^W,\\ &\qquad X_{t+1} = x_{t+1}\dots,X_T = x_T|X_t = i, y_t^1,\dots,y_t^W)\\ &= \sum_{\{x_{t+1},\dots,x_T\}\in\mathcal{M}^{T-t}} \mathbb{P}(y_{t+2}^1,\dots,y_T^1,\dots,y_{t+2}^W,\dots,y_T^W,\\ &\qquad X_{t+2} = x_{t+2}\dots,X_T = x_T|X_{t+1} = x_{t+1},\\ &\qquad X_t = i, y_t^1,\dots,y_t^W, y_{t+1}^1,\dots,y_{t+1}^W)\\ &\qquad \cdot\mathbb{P}(y_{t+1}^1|X_{t+1} = x_{t+1}, X_t = i, y_t^1,\dots,\\ &\qquad y_t^W)\cdots\mathbb{P}(y_{t+1}^W|X_{t+1} = x_{t+1}, X_t = i, y_t^1,\dots,\\ &\qquad y_t^W, y_{t+1}^1,\dots,y_{t+1}^{W-1})\cdot\mathbb{P}(X_{t+1} = x_{t+1}|X_t = i)\\ &= \sum_{j=1}^M \mathbb{P}(y_{t+2}^1,\dots,y_T^1,\dots,y_{t+2}^W,\dots,y_T^W,|X_{t+1} = j,\\ &\qquad y_{t+1}^1,\dots,y_{t+1}^W)\cdot\mathbb{P}(y_{t+1}^1|X_{t+1} = j, X_t = i, y_t^1,\\ &\qquad \dots,y_t^W)\cdots\mathbb{P}(y_{t+1}^W|X_{t+1} = j, X_t = i, y_t^1,\dots,\\ &\qquad y_t^W, y_{t+1}^1,\dots,y_{t+1}^{W-1})\cdot\mathbb{P}(X_{t+1} = j|X_t = i)\\ &= \sum_{j=1}^M A_{ij}[\prod_{w=1}^W L^w_{j,y_{t-1}^w y_t^w}]\beta_j(t+1).\end{aligned} \tag{15}$$

2) Some expressions in terms of forward-backward variables: In order to re-write (12) in terms of forward-backward equations, we first write the following in terms of the forward-backwards equations. Note for lack of space we are dropping the random variable Y term in front of the y terms.

$$\begin{aligned}\mathbb{P}(\mathbf{Y} = \mathbf{y}, X_t = i) &= \mathbb{P}(y_1^1,\dots,y_T^1,\dots,y_1^W,\dots,y_T^W,\\ &\qquad X_t = i)\\ &= \alpha_i(t)\beta_i(t).\end{aligned} \tag{16}$$

$$\begin{aligned}&\mathbb{P}(\mathbf{Y} = \mathbf{y}, X_{t-1} = i, X_t = j) = \mathbb{P}(y_1^1,\dots,y_T^1,\dots,y_1^W,\dots,\\ &\qquad y_T^W, X_{t-1} = i, X_t = j)\\ &= \mathbb{P}(y_t^1,\dots,y_T^1,\dots,y_t^W,\dots,y_T^W, X_t = j|y_1^1,\dots,\\ &\qquad y_{t-1}^1,\dots,y_1^W,\dots,y_{t-1}^W, X_{t-1} = i)\cdot\mathbb{P}(y_1^1,\dots,\\ &\qquad y_{t-1}^1,\dots,y_1^W,\dots,y_{t-1}^W, X_{t-1} = i)\\ &= \alpha_i(t-1)\cdot\mathbb{P}(y_t^1,\dots,y_T^1,\dots,y_t^W,\dots,y_T^W|y_1^1,\\ &\qquad \dots,y_{t-1}^1,\dots,y_1^W,\dots,y_{t-1}^W, X_{t-1} = i, X_t = j)\cdot\\ &\qquad \mathbb{P}(X_t = j|X_{t-1} = i)\\ &= \alpha_i(t-1)A_{ij}\cdot\mathbb{P}(y_{t+1}^1,\dots,y_T^1,\dots,y_t^W,\dots,y_T^W|\\ &\qquad y_1^1,\dots,y_t^1,\dots,y_1^W,\dots,y_{t-1}^W, X_{t-1} = i, X_t = j)\cdot\\ &\qquad \mathbb{P}(y_t^1|y_{t-1}^1, X_t = j)\\ &= \alpha_i(t-1)A_{ij}[\prod_{w=1}^W L^w_{j,y_{t-1}^w y_t^w}]\beta_j(t).\end{aligned} \tag{17}$$

3) Parameter estimates in terms of forward-backward variables: Using (16) and (17), the expressions in (12) in terms of forward-backward equations becomes,

$$\begin{cases}\widehat{\pi}_i = \frac{\alpha_i(1)\beta_i(1)}{\sum_{j=1}^M \alpha_i(1)\beta_i(1)},\\ \widehat{A}_{ij} = \frac{\sum_{t=2}^T \alpha_i(t-1)\widehat{A}'_{ij}\left[\prod_{w=1}^W \widehat{L}'^w_{j,y_{t-1}^w y_t^w}\right]\beta_j(t)}{\sum_{t=2}^T \alpha_i(t-1)\beta_i(t-1)},\\ \widehat{L}^{w,i}_{\mu\nu} = \frac{\sum_{t=2}^T \alpha_i(t)\beta_i(t)\mathbb{1}(Y_{t-1}^w=\mu)\mathbb{1}(Y_t^w=\nu)}{\sum_{t=2}^T \alpha_i(t)\beta_i(t)\mathbb{1}(Y_{t-1}^w=\mu)}.\end{cases} \tag{18}$$

Re-scaling forward backward equations: From (13) we see that as t increases the values of $\alpha_i(t)$ and $\beta_i(t)$ become very small. Hence, the terms in (18) diverge when computed numerically. To avoid this numerical problem we normalize the values of the forward and backward equations

$$\begin{cases} \bar{\alpha}_i(t) = \frac{\alpha_i(t)}{\sum_{i'=1}^{M} \alpha_{i'}(t)}, \\ \bar{\beta}_i(t) = \frac{\beta_i(t) \cdot \sum_{i'=1}^{M} \alpha_{i'}(t)}{\sum_{i'=1}^{M} \alpha_{i'}(T)}. \end{cases} \tag{19}$$

4) Stopping criterion for the EM algorithm: We enforce two simultaneous *stopping criteria* for the EM algorithm:

1) The number of iterations, n_i, exceeds a predefined threshold value, n_{max}, i.e., $n_i \geq n_{\text{max}}$.
2) The improvements in λ have reached a desired minimum, ϵ, i.e., $\Delta\lambda_{\text{min}} = \max_{n_m} |\lambda^{n_i} - \lambda^{n_i - n_m}| \leq \epsilon$, where λ^{n_i} is λ for the iteration n_i.

If any one of the stopping criteria is true the algorithm stops. We take $n_{\text{max}} = 10^4, n_m = 100$ and $\epsilon = 10^{-6}$ for our simulations.

IV. EDF Comparison Figure

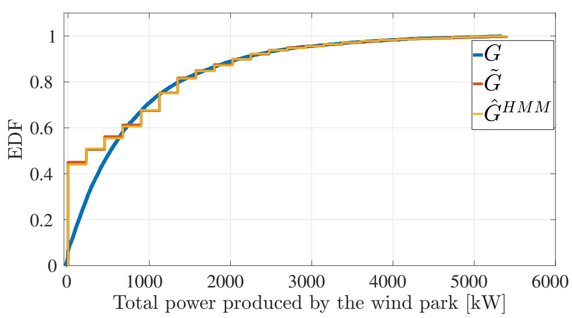

Fig. 3: Comparing EDFs of the total power produced by the wind farm for data sets P^w, $\tilde{P}^w$ and $\hat{P}^w_{\text{HMM}}$, $\forall w \in \mathcal{W}$ for $N = 5, M = 9$.

V. HMM complexity and improvements

The problem of inferring the probability of a state sequence, i.e., finding the stochastic state sequence (based on a MC or a HMM), which is most likely to explain a given observation sequence (time series of data), can be solved by using the well-known Viterbi algorithm based on dynamic programming, see [1] and the references therein. The problem of learning the underlying model (MC or HMM), i.e., the corresponding one step probability transition matrix, which are referred to as the model parameters, is solved by iteratively adjusting the model parameters to optimize the model fitting according to some criterium, e.g., the maximum-likelihood (ML). For HMMs, the iterative parameter estimation is typically performed using an expectation-maximization (EM) algorithm with the ML criterion. In the context of HMM, the EM algorithm becomes the Baum-Welch algorithm, [5]. Given a training data set, the EM algorithm iteratively estimates the HMM parameters in two stages; an expectation step (E-step) followed by a maximization step (M-step). In order to reduce the number of computations required for this process, one of the most common approaches proposed uses Viterbi training [2], [3].

The EM algorithm is an iterative scheme that is well-defined and is numerically stable, but convergence may require a large number of iterations, and may very well lead to local optima. Therefore, starting with a (any) set of parameters and iteratively re-estimating the parameters, one can improve the likelihood function until some limiting point is reached. For this reason, the choice of the starting set of parameters can be of crucial importance. Furthermore, the number of states should be specified in advance. In fact, learning the model topology is a difficult task, see [1] and the references therein. In [5], the authors describe the EM algorithm and they compare it with other relevant algorithmic approaches. Furthermore, they suggest some improvements and variations to deal with both the off-line and the on-line case. Finally, in the mathematical appendix of [5], the authors sketch the mathematical details of the algorithms.

A. HMMs with second order Markov properties

HMMs have been generalized to also cover "higher order Markov properties", see, e.g., [4] and the references therein. Thus, one could possibly extend the first order Markov models used in the paper to capture second or higher orders Markov properties. However, this would be out of the scope of the study we performed as the tail distribution is already very accurately captured and an extension to a higher order Markov property would result in an increase to the state-space HMM description and thus an increase in the computational complexity of the model.

VI. Parameter estimates for same one-step transition matrix model

In this case we assume that the individual wind turbine power output processes $\mathbf{Y}^w$, are governed by matrices (L^i) and A as follows:

$$\begin{cases} L^i_{\mu\nu} = \mathbb{P}\big(Y^w_{t+1} = \nu | Y^w_t = \mu, X_{t+1} = i\big) \\ A_{ij} = \mathbb{P}\big(X_{t+1} = j | X_t = i\big), \end{cases} \tag{20}$$

The initial distribution of Y^w_1 is given by $\rho^i_\mu = \mathbb{P}(Y^w_1 = \mu | X_1 = i)$. The parameter estimates for this model are given by

$$\begin{cases} \widehat{\pi}_i = \frac{\alpha_i(1)\beta_i(1)}{\sum_{j=1}^{M}\alpha_i(1)\beta_i(1)}, \\ \widehat{A}_{ij} = \frac{\sum\limits_{t=2}^{T}\alpha_i(t-1)\widehat{A}'_{ij}\left[\prod\limits_{w=1}^{W}\widehat{L}'^{j}_{y^w_{t-1}y^w_t}\right]\beta_j(t)}{\sum\limits_{t=2}^{T}\alpha_i(t-1)\beta_i(t-1)} \\ \widehat{\rho}^i_\mu = \frac{1}{W}\sum\limits_{w=1}^{W}\mathbb{1}(Y_1^w=\mu), \text{and} \\ \widehat{L}^i_{\mu\nu} = \frac{\sum\limits_{t=2}^{T}[\alpha_i(t)\beta_i(t)\sum\limits_{w=1}^{W}\mathbb{1}(Y^w_{t-1}=\mu)\mathbb{1}(Y^w_t=\nu)]}{\sum\limits_{t=2}^{T}[\alpha_i(t)\beta_i(t)\sum\limits_{w=1}^{W}\mathbb{1}(Y^w_{t-1}=\mu)]}. \end{cases} \tag{21}$$

The recursions of the forward-backward equations are given by $\forall w \in \mathcal{W},\ \forall i \in \mathcal{M}$

$$\begin{cases} \alpha_i(t) = \sum_{j=1}^{M}\alpha_j(t-1)A_{ji}[\prod\limits_{w=1}^{W}L^i_{y^w_{t-1}y^w_t}], \\ \beta_i(t) = \sum_{j=1}^{M}A_{ij}[\prod\limits_{w=1}^{W}L^j_{y^w_{t-1}y^w_t}]\beta_j(t+1), \end{cases} \tag{22}$$

with initialization $\alpha_i(1) = \pi_i \cdot \prod\limits_{w=1}^{W}\rho^i_{y^w_1}$ and $\beta_i(T) = 1$.

VII. Comparison Tables

In Table I we compare $\bar{\gamma}^{(N,M)}_{\hat{P}_{\text{HMM}}}$ for different quantile thresholds, G^*, with γ_P for $N = 5$ and different number of hidden states M for the model described in *(3) of the main text*. We tabulate the relative error of $\bar{\gamma}^{(N,M)}_{\hat{P}_{\text{HMM}}}$ and γ_P, RE $= \frac{|\gamma_P - \bar{\gamma}^{(N,M)}_{\hat{P}_{\text{HMM}}}|}{\gamma_P} \times 100$ (in %). We also tabule $-\log\widehat{\mathcal{L}}$, AIC values, BIC values and the number of parameters in the model p in the table in order to find the best fitted model.

In Table II we compare $\gamma'^{(N,M)}_{\hat{P}_{\text{HMM}}}$ for different quantile thresholds, G^*, with γ_P for $N = 5$ and different number of hidden states M for the model described in *Section IV-B of the main text* and by (20). We tabulate the relative error RE $= \frac{|\gamma_P - \gamma'^{(N,M)}_{\hat{P}_{\text{HMM}}}|}{\gamma_P} \times 100$ (in %), $-\log\widehat{\mathcal{L}}$, AIC & BIC values and the number of parameters in the model p in the table.

In Table III we compare the quantile fraction values for different number of hidden states M for the macroscopic approach described in *Section V of main text*. Given the observation of the binned measurement total power produced by the wind farm data $\hat{G} = [\hat{G}_1, \ldots, \hat{G}_T]$ we first estimate the the HMM model parameters. We generate surrogate data series of the total power produced by the wind farm, $\bar{G}$ from the model parameters. $\gamma^{(N,M)}_{\bar{G}} = \frac{1}{T}\sum_{t=1}^{T}\mathbb{1}(\bar{G}_t > G^*)$ is the fraction of time the total power produced by the wind farm obtained from HMM is greater than G^* threshold. We generate 100 realizations of $\bar{G}$ of length $T = 10^4$. Then we calculate the mean and the standard deviations of $\gamma^{(N,M)}_{\bar{G}}$ from the 100 realizations, $\bar{\gamma}^{(N,M)}_{\bar{G}}$ and $\sigma(\gamma^{(N,M)}_{\bar{G}})$ respectively.

We also tabulate the relative error $RE = \frac{|\bar{\gamma}^{(N,M)}_{\bar{G}} - \gamma_G|}{\gamma_G} \times 100$ (in %). In order to find the best model fit we also compare the values of $-\log\widehat{\mathcal{L}}$, AIC and BIC. Note that the simplest discrete time Markov model, i.e. $M = 1$, can capture the quantile fractions very well, however the values of $-\log\widehat{\mathcal{L}}$, AIC and BIC are much higher compared to the $M \geq 2$ cases.

References

[1] E. Messina and D. Toscani, "Hidden Markov models for scenario generation," *IMA Journal of Management Mathematics*, vol.19, no.4, pp. 379–401, 2008.

[2] C. J. Wellekens, "Mixture density estimators in Viterbi training," In 1992 IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP-92. IEEE, vol. 1, pp. 361–364, 1992.

[3] Y. Gotoh, M. M. Hochberg and H. F. Silverman. "Efficient training algorithms for HMMs using incremental estimation," *IEEE Transactions on Speech and Audio Processing*, vol.6, no. 6, pp. 539–548, 1998.

[4] J. Park, M. Chebbah, S. Jendoubi and A. Martin, "Second-order belief hidden Markov models," In International Conference on Belief Functions. Springer, pp. 284–293, 2014.

[5] P. Baldi and Y. Chauvin. "Smooth on-line learning algorithms for hidden Markov models," *Neural Computation*, vol. 6, no. 2, pp. 307–318, 1994.

TABLE I: Comparing $\gamma^{(N,M)}_{\hat{P}_{\text{HMM}}}$ for different quantile thresholds, G^*, with γ_P for $N = 5$ for different number of hidden states M.

M	Quantile	$\bar{\gamma}^{(5,M)}_{\hat{P}_{\text{HMM}}} \pm \sigma(\gamma^{(5,M)}_{\hat{P}_{\text{HMM}}})$	RE (%)	$-\log\widehat{\mathcal{L}} \times 10^{-4}$	AIC $\times 10^{-4}$	BIC $\times 10^{-4}$	p
1	90% 95% 99%	$6.67 \times 10^{-3} \pm 4.23 \times 10^{-5}$ $3.01 \times 10^{-4} \pm 7.12 \times 10^{-6}$ $1.70 \times 10^{-8} \pm 2.14 \times 10^{-8}$	93.33 99.39 99.99	3.4465	6.9218	7.0236	140
5	90% 95% 99%	0.1248 ± 0.0048 0.0573 ± 0.0028 0.0069 ± 0.0004	24.856 14.465 30.850	2.5176	5.1592	5.5974	620
6	90% 95% 99%	0.1022 ± 0.0031 0.0470 ± 0.0018 0.0035 ± 0.0002	2.2513 6.1418 65.270	2.4661	5.0822	5.6123	750
7	90% 95% 99%	0.0993 ± 0.0030 0.0468 ± 0.0020 0.0045 ± 0.0004	0.6857 6.5061 55.421	2.4534	5.1179	5.643	1056
8	90% 95% 99%	0.1013 ± 0.0036 0.0547 ± 0.0023 0.0082 ± 0.0005	1.3336 9.3667 18.232	2.4193	5.8016	5.9404	1215
9	90% 95% 99%	0.1011 ± 0.0032 0.0517 ± 0.0020 0.0114 ± 0.0060	1.1512 3.2605 13.127	2.4011	5.0776	6.0500	1376
10	90% 95% 99%	0.1268 ± 0.0050 0.0633 ± 0.0028 0.0076 ± 0.0005	26.887 26.496 24.654	2.4117	5.1312	6.2190	1539

TABLE II: Comparing $\gamma'^{(N,M)}_{\hat{P}_{\text{HMM}}}$ for different quantile thresholds, G^*, with γ_P for $N = 5$ for different number of hidden states M along with $-\log\widehat{\mathcal{L}}$, AIC and BIC values.

M	Quantile	$\bar{\gamma}'^{(5,M)}_{\hat{P}_{\text{HMM}}} \pm \sigma(\gamma'^{(5,M)}_{\hat{P}_{\text{HMM}}})$	RE (%)	$-\log\widehat{\mathcal{L}} \times 10^{-4}$	AIC $\times 10^{-4}$	BIC $\times 10^{-4}$	p
5	90% 95% 99%	0.0924 ± 0.0027 0.0477 ± 0.0017 0.0048 ± 0.0003	7.532 4.613 52.05	2.5421	5.1081	5.1929	120
6	90% 95% 99%	0.0983 ± 0.0030 0.0494 ± 0.0019 0.0048 ± 0.0004	1.707 1.290 52.17	2.5087	5.0975	5.1535	150
7	90% 95% 99%	0.1009 ± 0.0031 0.0497 ± 0.0018 0.0075 ± 0.0006	0.940 0.689 25.24	2.4936	5.0304	5.1830	216
8	90% 95% 99%	0.1025 ± 0.0029 0.0532 ± 0.0019 0.0126 ± 0.0009	2.532 6.369 25.45	2.4766	5.0041	5.1844	255
9	90% 95% 99%	0.0987 ± 0.0032 0.0517 ± 0.0020 0.0108 ± 0.0060	1.243 3.391 8.116	2.4724	5.0040	5.2132	296
10	90% 95% 99%	0.0984 ± 0.0043 0.0522 ± 0.0027 0.0076 ± 0.0008	1.562 4.313 1.728	2.4729	5.0038	5.2087	339

TABLE III: Comparing $\gamma_{\bar{G}}^{(N,M)}$ for different quantile thresholds, G^*, with γ_P for $N = 15$ for different number of hidden states M along with $-\log \widehat{\mathcal{L}}$, AIC, BIC and number of parameters of the model, p.

M	Quantile	$\bar{\gamma}_{\bar{G}}^{(15,M)} \pm \sigma(\gamma_{\bar{G}}^{(15,M)})$	RE (%)	$-\log \widehat{\mathcal{L}} \times 10^{-4}$	AIC $\times 10^{-5}$	BIC $\times 10^{-5}$	p
1	90%	0.0984 ± 0.0042	1.569	6.2738	1.2590	1.2738	210
	95%	0.0469 ± 0.0025	6.302				
	99%	0.0096 ± 0.0008	4.216				
2	90%	0.1067 ± 0.0101	6.717	1.0129	0.2116	0.2435	451
	95%	0.0509 ± 0.0060	1.788				
	99%	0.0110 ± 0.0024	9.918				
3	90%	0.1055 ± 0.0111	5.572	0.9983	0.2132	0.2613	680
	95%	0.0512 ± 0.0073	2.409				
	99%	0.0107 ± 0.0021	6.488				
4	90%	0.1173 ± 0.0128	17.37	0.9852	0.2152	0.2796	911
	95%	0.0567 ± 0.0076	13.34				
	99%	0.0101 ± 0.0021	0.487				
5	90%	0.0988 ± 0.0102	1.154	0.9761	0.2181	0.2989	1144
	95%	0.0475 ± 0.0057	5.095				
	99%	0.0101 ± 0.0022	0.507				
6	90%	0.1004 ± 0.0113	0.387	0.9793	0.2234	0.3209	1379
	95%	0.0471 ± 0.0065	5.850				
	99%	0.0096 ± 0.0020	4.597				
7	90%	0.1026 ± 0.0099	2.594	0.9708	0.2265	0.3407	1616
	95%	0.0505 ± 0.0063	0.866				
	99%	0.0113 ± 0.0025	13.07				