Advances in Computational Intelligence: 12th International Work-Conference on Artificial Neural Networks, IWANN 2013, Puerto de la Cruz, Tenerife, Spain, June 12-14, 2013, Proceedings, Part I. Lecture Notes in Computer Science, Volumen 7902. Springer, 2013. 106-113.

DOI: http://dx.doi.org/10.1007/978-3-642-38679-4_9

Diffusion Methods for Wind Power Ramp Detection

Ángela Fernández, Carlos M. Alaíz, Ana M. González,
Julia Díaz and José R. Dorronsoro

Departamento de Ingeniería Informática and Instituto de Ingeniería del Conocimiento
Universidad Autónoma de Madrid, 28049, Madrid, Spain
{a.fernandez,carlos.alaiz,ana.marcos,julia.diaz,jose.dorronsoro}@uam.es

Abstract. The prediction and management of wind power ramps is currently receiving large attention as it is a crucial issue for both system operators and wind farm managers. However, this is still a problem far from being solved and in this work we will address it as a classification problem working with delay vectors of the wind power time series and applying local Mahalanobis K-NN search with metrics derived from Anisotropic Diffusion methods. The resulting procedures clearly outperform a random baseline method and yield good sensitivity but more work is needed to improve on specificity and, hence, precision.

Keywords: Diffusion Methods, Anisotropic Diffusion, diffusion distance, wind power ramps.

1 Introduction

The growing presence of wind energy is raising many issues in the operation of electrical systems, some of them can be conceivably addressed through the application of Machine Learning (ML) techniques. One important example that we shall deal with in this paper is the prediction of wind ramps, i.e., sudden, large increases or decreases of wind energy production over a limited time period [8]. In fact, algorithms to detect possible ramps and raise alerts about them are of obvious interest to system operators and wind farm managers to support wind farm control, to decide how much energy should be dispatched or to modify generation schedules. However, there are still few methodologies for ramp prediction and even there is not a standard ramp definition yet, making this topic a wide open research area.

From a ML point of view, two approaches to wind ramp detection with different final goals have been proposed in the literature. If we want to determine not only the starting of a ramp but also its magnitude, regression models are the natural choice. This approach has been followed, for example, in [11], that applies multivariate time series prediction models and uses mean absolute prediction error and standard deviation as accuracy measures. In [3] probabilistic numerical weather prediction systems are used to associate uncertainty estimates to wind energy predictions and to provide risk indices to warn about possible large deviations and ramp events. On the other hand, we can pursue a classification-based approach to predict wind ramps through event detection models. Examples of this are [4] or [7], that try to locate ramp presence some hours before or after the wind energy forecasts, given some time into the future.

In this work we will also consider ramp detection as a classification problem but we will seek to provide for each hour a forecast on whether a ramp is about to start, which differs from the few available state-of-the-art results and makes them non comparable. Our overall approach is to relate the conditions at a given hour to similar conditions in the past and to somehow derive a ramp forecast from what happened in these previous similar situations. More precisely, we can consider for each hour t a certain feature vector X_t that should adequately represent wind energy behavior up to time t and find a subset of K past vectors X_{t_i} close to X_t in an appropriate metric. There is a growing number of options to choose data that characterize X_t but in this work we shall simply consider the wind energy production time series as the only such information and X_t will be a delay vector built from the last D wind energy production values, $X_t = (p_{t-D+1}, \dots, p_{t-1}, p_t)^T$. This is certainly not an optimal choice, as the well-known chaotic behavior of the weather implies that past behavior of wind energy up to time t has only a weak influence on its behavior after t. However, ramps are also local phenomena and, in any case, our approach can easily accommodate the use of more relevant information. For instance, the quality of wind energy prediction is steadily improving and can easily be incorporated to the procedures pursued here.

Going back to our approach, the most relevant issue is the definition of the metric to be used to compare X_t with previous values. To do so, we will work in an Anisotropic Diffusion context. In general, diffusion methods assume that sample values, the D-dimensional delay vectors X_t in our case, lie in a manifold $\mathcal{M}$ whose geometry corresponds to a diffusion distance associated with a Markov process. Then, the relationship between the spectral properties of the Markov chain and the manifold geometry allows the definition of a Diffusion Map into a lower dimensional space in such a way that Euclidean distance in the projected space corresponds to the diffusion metric on $\mathcal{M}$. However, this requires a computationally very costly eigenanalysis of the Markov transition matrix and we will pursue here an alternative Anisotropic Diffusion model which assumes that the sample data points are the result of the application of an unknown map f to the latent variables l_t that govern the X_t data and that follow a particular independent stochastic Itô process. This allows to estimate the Euclidean distance in the inaccessible l space through local Mahalanobis distances in the sample manifold $\mathcal{M}$ *without* having to go through any costly eigenanalysis.

Wind power clearly has a time structure and if we assume weather and, thus, wind power as governed by a latent variable model, the wind ramp detection problem fits nicely in the Anisotropic Diffusion framework. In this paper we will explore this approach and, as we shall see, our methods clearly improve on a baseline random model and have good sensitivity. However, specificity and, hence, precision must be improved. Moreover, while slightly better, the Mahalanobis models still give results similar to those achieved using a simple Euclidean metric. Still, there is a clear room from improvement. In fact, it is well known that delay vectors are a too crude representation of the wind power time series and that they cannot be used, for instance, for forecasting future power values. We will briefly discuss this at the end of the paper, that is organized as follows. In Sect. 2 the diffusion theory framework is introduced and in Sect. 3 the wind ramp detection problem is presented and our prediction methods proposed. Sect. 4 contains the numerical experiments and, finally, Sect. 5 ends this paper with a brief discussion and some conclusions and hints for further work.

2 Diffusion Methods Review

We give first a simplified review of standard Diffusion Methods (DMs) following the notation of [10]. The first step is to build a complete connectivity graph G where the original points are the graph nodes and where the weight distances reflect the local similarity between two points X_i, X_j, i.e., we have $w_{i,j} = W_\sigma(X_i, X_j) = h\left(\rho(X_i, X_j)^2/\sigma\right)$, where h is a function with exponential decay, such as a Gaussian kernel, ρ is some metric and σ is a parameter that defines the "locality" of a neighborhood. Weights are then normalized as $K = D^{-1}W$, with $D_{ii} = \sum_j w_{ij}$ the graph degree (D is a diagonal matrix). K is then a Markov matrix that can be iterated to generate a Markov process with transition probabilities $P_t(X_i, X_j)$. This can in turn be used to define the spectral distance

$$\mathcal{D}_t(X_i, X_j) = \|P_t(X_i, \cdot) - P_t(X_j, \cdot)\|^2_{L_2} = \sqrt{\sum_k |P_t(X_i, X_k) - P_t(X_j, X_k)|^2} \ ,$$

that express the similarity after t steps between two diffusion processes starting from X_i and X_j. While it is rather hard to compute this distance, it turns out that the eigenfunctions $\{\Psi_i\}$ of the operator K coincide with the eigenfunctions $\{\Phi_i\}$ of the graph Laplacian (see [2, 5]), which is defined as

$$\mathcal{L} = D^{-\frac{1}{2}} W D^{-\frac{1}{2}} - I = D^{\frac{1}{2}} K D^{-\frac{1}{2}} - I \ .$$

This can be used to show that $\mathcal{D}_t(X_i, X_j)$ coincides with Euclidean distance in the DM space.

The study of DMs has opened a world of possibilities in dimensionality reduction [5], clustering [2] or function approximation [10]. However, the eigenanalysis needed to compute the DMs is still quite costly computationally and, moreover, their application to new patterns is not straighforward and requires the use of a Nyström approximation.

We will focus our attention here on the anisotropic version of these methods [9], which fits nicely to the problem we want to solve. The starting point is to assume that the sample is generated by a non linear function f acting on some d-dimensional parametric features l_t that follow an Itô process

$$dl^j = a^j(l)dt + b^j(l)dw^j, \ j = 1, \ldots, d,$$

where a^j is the drift coefficient, b^j is the noise coefficient and w^j is a Brownian motion. Itô's Lemma ensures that our observable variables $X_t = f(l_t)$ are also Itô processes. Thanks to this fact, and assuming an appropriate feature rescaling, we can locally estimate the distortion in the transformation f through the covariance matrix C of the observable data, namely $C = JJ^T$, where J is the Jacobian of the function f. The important fact now is that the Euclidean distance $\|l_i - l_j\|$ in the latent variable space can be approximated as

$$||l_i - l_j||^2 \simeq (X_i - X_j)^T [C^{-1}(X_i) + C^{-1}(X_j)](X_i - X_j). \tag{1}$$

We can now build a diffusion kernel based on this distance whose infinitesimal generator coincides with a backward Fokker–Planck operator. In particular, the original latent features could be recovered by the appropriate eigenanalysis of this operator. However, we do not need this to estimate distances in the inaccessible latent space as they can be computed directly on the sample manifold $\mathcal{M}$ using (1) *without* having to go through any costly eigenanalysis.

3 Predicting Wind Ramps

In this section we will give a proposal for wind ramp warnings. As mentioned before, while the idea of a wind ramp is intuitively clear, there is no universally accepted characterization of it. Thus, here we shall discuss first the definition of wind ramps, present then an approach for issuing wind ramp warnings and close this section with the methodology which we will use to evaluate its effectiveness.

As mentioned in [6], an intuitive description of a wind power ramp could be a large change in wind production in a relatively short period of time. To turn this description into a formal definition we need to specify what are a "large change" and a "short time period". Several options are discussed in [3, 6] but possibly the simplest one is to consider derivatives or, rather, first order differences, and say that a ramp will happen at time t if in a time period Δt we have

$$|P(t + \Delta t) - P(t)| > \Delta P_{th} \ .$$

Notice that this definition detects equally upward and downward ramps and it requires to determine the values of Δt and the threshold ΔP_{th}. Starting with Δt, if t is given in hours, a low value such as $\Delta t = 1$ leaves no reaction time to the system operator; on the other hand, a larger value will not imply a big impact on the electrical system. Because of these and similar considerations (see [7]), we have settled on the value $\Delta t = 3$. Notice that once Δt is chosen, ΔP_{th} essentially determines how often ramps happen. A low threshold results in many ramp events but most of them will be of little consequence, while large values result very relevant but also very infrequent ramps. We have settled in a ΔP_{th} that marks the top 5% percentile of ramp events. In other words, the probability of a ramp jump $|P(t + \Delta t) - P(t)|$ larger than ΔP_{th} is 0.05.

In order to apply Anisotropic Diffusion to ramp event prediction, we have to assume that extreme power ramps correspond to particular values of the unknown latent variables that determine wind energy production. More precisely, we have to define wind energy patterns X_t that somehow capture the structure of wind production at time t and that are determined by latent variable values l_t. Thus, a possible approach to predict ramps at time t is to identify previous latent vectors l_{t_i} that are close to the current latent vector l_t and to exploit the corresponding previous wind energy patterns X_{t_i} to deduce whether the current pattern X_t is associated to a ramp event. To make this work, we must have an estimate of the distance $\|l_{t_i} - l_t\|$ and is in this context where we can benefit of an Anisotropic Diffusion approach. As explained in Sect. 2, this framework allows to approximate $\|l_{t_i} - l_t\|$ by (1). This estimate requires to compute and invert the covariances $C(X_{t_i})$ at each possible X_{t_i}. To alleviate the possibly large computational cost, we simplify the Mahalanobis distance to $d(X_{t_i}, X_t) = (X_{t_i} - X_t)^T C_t^{-1} (X_{t_i} - X_t)$, with C_t^{-1} the inverse of the local covariance matrix in a cloud of points around X_t.

We shall apply this approach working with D-dimensional energy patterns of the form $X_t = (p_{t-D+1}, \ldots, p_{t-1}, p_t)^T$ that correspond to a delay window of length D, for which we will find a subset $\mathcal{S}_t$ with the K sample patterns X_{t_i} nearest to X_t, with K appropriately selected. This will be done for both the Mahalanobis and the Euclidean (i.e., isotropic) distances. Once $\mathcal{S}_t$ is found, we will classify X_t as a ramp if we have $\nu_t \geq \rho$, with ν_t the number of ramp-associated patterns in $\mathcal{S}_t$ and $1 \leq \rho \leq K$ a threshold value; we will give results only for $\rho = 1$ but larger ρ values would be associated to more confidence in a ramp happening at time t. In the Mahalanobis case we also have to select a pattern cloud $\mathcal{C}_t$ to compute the covariance matrix C_t at time t. The simplest way is just to work with a *time cloud*, i.e., to select $\mathcal{C}_t = \{X_t, X_{t-1}, \ldots, X_{t-M+1}\}$, using the M patterns closest to X_t in time. Alternatively, we shall consider a *cluster cloud* where we fix a larger time cloud with κM patterns, apply κ-means clustering to it and choose the new cloud $\mathcal{C}_t^\kappa$ as the cluster that contains X_t. Besides the parameter ρ, that will affect the confidence on the ramp prediction, performance will of course depend on the concrete selection of the parameters used, namely the number K of patterns closest to X_t, the dimension D of the patterns, and the M and κ used to determine the covariance cloud. The complete method is summarized in Alg. 1.

Since we want to solve what essentially is a supervised classification problem, confusion matrix-related indices seem to be the best way to evaluate algorithm performance. More precisely, we use the sensitivity $\text{Sens} = {}^{\text{TP}}/_{\text{TP}+\text{FN}}$ and specificity $\text{Spec} = {}^{\text{TN}}/_{\text{TN}+\text{FP}}$ values, as well as precision $\text{Prec} = {}^{\text{TP}}/_{\text{TP}+\text{FP}}$, that measures the proportion of correct ramp alerts. In order to select the best K, D, M and κ values we will combine $\text{TP}, \text{TN}, \text{FP}$ and FN in the Matthews correlation coefficient [1]

$$\Phi = \frac{\text{TP} \cdot \text{TN} - \text{FP} \cdot \text{FN}}{\sqrt{(\text{TP}+\text{FP}) \cdot (\text{TN}+\text{FN}) \cdot (\text{TP}+\text{FN}) \cdot (\text{TN}+\text{FP})}}$$

that returns a $[-1, 1]$ value with $\Phi = 1$ if $\text{FP} = \text{FN} = 0$, i.e., when we have a diagonal confusion matrix, and $\Phi = -1$ if $\text{TP} = \text{TN} = 0$.

Algorithm 1 Ramp Events Detection

Input: $\boldsymbol{P} = \{p_1, \ldots, p_s\}$, wind power time series; $\boldsymbol{D}$, pattern dimension; $\boldsymbol{\Delta t}$, ramp duration; $\boldsymbol{R} = \{r_1, \ldots, r_{s-\Delta t}\}$, $\{0, 1\}$ ramp labels; $\boldsymbol{\rho}$, ramp threshold.

Output: $\hat{\boldsymbol{r}}_{\boldsymbol{s+1}}$, the ramp prediction at time $s + 1$.

Build patterns $X_t = (p_{t-D+1}, \ldots, p_{t-1}, p_t)^T$;
Select the cloud $\mathcal{C}_s$;
Compute the covariance C_s;
$\mathcal{S}_s = NearestNeightbors(X_s, C_s, K)$, the K patterns closest to X_s;
$\nu_s = \sum_{\mathcal{S}_s} r_{s_i}$;
if $\nu_s \geq \rho$ **then**
  return 1;
else
  return 0;
end if

4 Experiments

In this section we will illustrate the application of the previous methods to the Sotavento wind farm[1], located in the northern Spanish region of Galicia and that makes its data publicly available. The training data set that we will use is composed of hourly productions from July 1, 2010, to July 31, 2012. Of these, each hour starting from August 1 2011 will be used for test purposes, with the training set formed by one year and one month, up to the hour before. Wind ramp hours have been defined as those hours h for which the absolute 3-hourly difference between productions at hours h and $h + 4$ falls in the top 5%. This means a power rise of at least 4.38MW, which essentially correspond to a 25% of the nominal power of this wind farm, a value also used in other studies [3, 4]. We recall that straight wind ramp prediction is a rather difficult problem for which there are not reference results in the literature. Thus we will use as a baseline reference the performance of a random prediction that assigns at each hour a ramp start with a 0.05 probability. For the test period considered, we have $N = 8699$ patterns of which 5%, i.e. $N_p = 450$, are ramps and the rest, $N_n = 8249$, are not. Table 1 shows the expected values of the confusion matrix of this random model as well as the mean and deviation of sensitivity, specificity and precision. We will compare the baseline results with the three K-nearest neighbors (NN) models previously considered, that is, standard Euclidean K-NN, called NN^E, and Mahalanobis K-NN with either a time cloud covariance, called NN^M_T, or a cluster cloud, called NN^M_C. We have to appropriately set the hyper-parameter values, namely pattern dimension D and time cloud size M. To arrive to some good values of K, D and M we have considered K values in $\{5, 10, 15\}$, D values in $\{4, 8, 12\}$ and M values in $\{10, 20, 50\}$ (we fix $\kappa = 4$ to define clouds in the cluster approximation) and chosen as the best parameters those giving a largest Matthews coefficient Φ, which are $K = 15$ and $D = 4$ for all cases, and $M = 50$ and $\kappa M = 200$ for the time and cluster cloud sizes.

[1] Sotavento Galicia, `http://www.sotaventogalicia.com/index.php`.

Table 1: Baseline model confusion matrix.

	Pred. +	Pred. −	$\sum$		Mean	Deviation
Real +	181	269	450	Sens	40.13%	2.31%
Real −	3310	4939	8249	Spec	59.87%	0.54%
$\sum$	3491	5208	8699	Prec	5.17%	0.30%

Table 2: K-NN models confusion matrices.

Time Cloud (NN_T^M)

	P. +	P. −	$\sum$
R. +	321	129	450
R. −	3048	5201	8249
$\sum$	3369	5330	8699
Sens	71.33%		
Spec	63.05%		
Prec	9.53%		

Cluster Cloud (NN_C^M)

	P. +	P. −	$\sum$
R. +	318	132	450
R. −	3001	5248	8249
$\sum$	3319	5380	8699
Sens	70.67%		
Spec	63.62%		
Prec	9.58%		

Euclidean (NN^E)

	P. +	P. −	$\sum$
R. +	314	136	450
R. −	3034	5215	8249
$\sum$	3348	5351	8699
Sens	69.78%		
Spec	63.22%		
Prec	9.38%		

The results obtained with each optimal model are presented in Table 2. As it can be seen, all K-NN methods clearly outperform the random baseline model, as the sensitivity and specificity of any random predictor always sum 100%. The improvement can be appreciate particularly with respect to sensitivity, that goes from near 40% to about 70%, and precision, that goes from near 5% to about 9.5%. The specificity gain is smaller, about 4%, but still quite larger than the 0.54% standard deviation of the random model. On the other hand, the NN_T^M and NN_C^M models are only slightly better than the purely Euclidean model NN^E and none of the methods can be considered as exploitation-ready models, for while they give a good sensitivity, but specificity and, therefore, precision are far from good enough. Nevertheless, as we discuss next, it is well known that delay vectors provide a rather crude information about the wind power time series and adding more information to the X_t is a clear first step toward better wind ramp detection.

5 Discussion and Conclusions

While they are a key problem in wind energy and system operation management, there is still no standard definition of wind power ramps and their detection is therefore a problem far from being solved. In this work we have applied an Anisotropic Diffusion approach where we consider wind power delay vectors as visible events derived from latent vectors that follow some Itô processes. This leads naturally to define a covariance based Mahalanobis distance for the delay vectors and, in turn, to apply K-NN methods to detect past vectors close to the current one X_t and to use this information to predict whether or not a ramp is going to start at time t. The resulting methods clearly outperform a baseline random model and show a good sensitivity. However, specificity must be improved which, in turn, would lead to better precision and, hence, to systems ready to industrial use. A first step to achieve this would be to refine ramp prediction using some weighted combination of the ramp states of the K nearest neighbors of X_t. A second step would be to work with patterns X_t richer than plain delay vectors, adding for instance numerical weather prediction (NWP) information or even short time wind power predictions derived from this NWP information. Finally, we could also exploit the time evolution of previous wind ramp alerts to improve specificity. We are working on these and similar directions.

Acknowledgement. With partial support from Spain's grant TIN2010-21575-C02-01 and the UAM–ADIC Chair for Machine Learning. The first author is also supported by an FPI–UAM grant and kindly thanks the Applied Mathematics Department of Yale University for receiving her during her visits. The second author is supported by the FPU–MEC grant AP2008-00167.

References

1. Baldi, P., Brunak, S., Chauvin, Y., Andersen, C., Nielsen, H.: Assessing the Accuracy of Prediction Algorithms for Classification: An Overview. Bioinformatics 16, 412–424 (2000)
2. Belkin, M., Nyogi, P.: Laplacian Eigenmaps for dimensionality reduction and data representation. Neural Computation 15(6), 1373–1396 (2003)
3. Bossavy, A., Girard, R., Kariniotakis, G.: Forecasting ramps of wind power production with numerical weather prediction ensembles. Wind Energy 16(1), 51–63 (2013)
4. Bradford, K., Carpenter, R., Shaw, B.: Forecasting southern plains wind ramp events using the wrf model at 3-km. In: Proceedings of the AMS Student Conference. Atlanta, Georgia (2010)
5. Coifman, R., Lafon, S.: Diffusion Maps. Applied and Computational Harmonic Analysis 21(1), 5–30 (2006)
6. Ferreira, C., Gama, J., Matias, L., Botterud, A., Wang, J.: A survey on wind power ramp forecasting. Tech. rep., Argonne National Laboratory (February 2011)
7. Greaves, B., Collins, J., Parkes, J., Tindal, A.: Temporal forecast uncertainty for ramp events. Wind Engineering 33(4), 309–319 (2009)
8. Kamath, C.: Associating weather conditions with ramp events in wind power generation. In: Power Systems Conference and Exposition (PSCE), 2011 IEEE/PES. pp. 1–8 (march 2011)
9. Singer, A., Coifman, R.: Non-linear independent component analysis with Diffusion Maps. Applied and Computational Harmonic Analysis 25(2), 226–239 (2008)
10. Szlam, A., Maggioni, M., Coifman, R.: Regularization on graphs with function–adapted diffusion processes. Journal of Machine Learning Research 9, 1711–1739 (2008)
11. Zheng, H., Kusiak, A.: Prediction of wind farm power ramp rates: A data-mining approach. Journal of Solar Energy Engineering 131(3), 031011–1–031011–8 (2009)