Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 52-11 (2019) 110–115

Handling Big Datasets in Gaussian Processes for Statistical Wind Vector Prediction [⋆]

Matija Perne [*] **Martin Stepančič** [*] **Boštjan Grašič** [**]

[*] *Jožef Stefan Institute, Jamova cesta 39, 1000 Ljubljana, Slovenia*
[**] *MEIS d.o.o., Mali Vrh pri Šmarju 78, 1293 Šmarje - Sap, Slovenia*

Abstract: We construct a statistical model predicting wind vector over very complex terrain characterized by low wind speeds and changeable wind directions. These are necessary inputs for atmospheric dispersion modelling of hypothetical radioactive pollution events in short-term or medium-term future to better protect the local population. The statistical model uses predictions of a numerical weather prediction model as some of its inputs, so they together form a hybrid model. The statistical model is realized as a nonlinear autoregressive exogenous model whose dynamics is described with a Gaussian process model. It relies on training data, and there is more training data available than the computing system is able to process. One possibility of avoiding this issue is to use a randomly selected subset of the available historical measurements as the training data. However, a better choice of training data may result in a model that performs better. We develop and test a smart training set selection method that selects the training data points based on Euclidean distances between them. The resulting model improvement is insignificant and inconsistent. We explore the reasons for underperformance of the method. We conclude that our example does not offer much opportunity for training set selection methods to achieve better results than random selection.

Keywords: Gaussian processes, Wind speeds, Modelling errors, Statistics, Autoregressive models, Probabilistic models

1. INTRODUCTION

The motivation for the study is atmospheric dispersion modelling of a hypothetical unplanned radioactive emission in short-term and medium-term future. The pollutant would be emitted from Krško Nuclear Power Plant (NPP), Slovenia, and modelling results could be used to protect the local population in such an event. There are already existing facilities for this purpose in place. In the region of interest, there are several weather stations and a numerical weather prediction (NWP) system is in operation (Božnar et al., 2012). We use these resources with the intention of improving on them. We focus on wind speed and direction.

The atmospheric dispersion model requires local meteorological variables as its inputs. They can be either measured or predicted, and when modelling atmospheric dispersion for the future, only the latter option is available. Models for predicting the local meteorological variables are thus necessary.

There are various kinds of models for weather prediction. NWP is based on physical understanding of the system. However, it has low temporal and spatial resolution. It cannot take into account detailed topography and land use information, and its predictions are for cell averages, not for local meteorological variables at a meteorological station. At the other extreme, there is persistence method that uses the current measurement as forecast and gives good short-term results on flat terrain (Potter and Negnevitsky, 2006). There are also more sophisticated statistical black box models (Wiener, 1961) that identify patterns in training data and use them to make predictions for the future from available inputs such as current measurements. These models by definition do not utilize explicit knowledge about the underlying mechanisms. In our case, the statistical model for predicting ground level wind at a certain location uses some outputs of the NWP model for its inputs, forming a hybrid model.

For the statistical part of the hybrid model, we use Gaussian process (GP) modelling. The main reason for using GP is that it provides information on output uncertainty (Kocijan, 2016). The GP model is typically nonparametric and has the property of the universal approximator (of any square-integrable function). However, the computational burden of the GP model identification increases with the number of measurements. The issue could be solved up to an extent with the sparse GP modelling methods or Evolving GP models (Kocijan, 2016; Petelin et al., 2015; Kocijan et al., 2005). All sparse approximate methods try to retain the bulk of the information contained in the full training dataset, but reduce the size of the numerical matrices to facilitate a less computationally demanding implementation of the GP model (Petelin et al., 2013). Instead of trying to solve the problem of infeasibly big

[⋆] The authors acknowledge the project “Method for the forecasting of local radiological pollution of atmosphere using Gaussian process models”, ID L2-278174, and research core funding No. P2-0001, which were financially supported by the Slovenian Research Agency.

2405-8963 © 2019, IFAC (International Federation of Automatic Control) Hosting by Elsevier Ltd. All rights reserved.
Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2019.09.126

model dataset in a general way, we are concerned with the influence of the choice of regressors (statistical model inputs) and training data set on the model performance in the case of statistical weather prediction. In our case, we study the consequences of reducing the set of measurements in two specific ways: randomly picking the data and selecting based on the Euclidean distance between datapoints.

2. METHODS

We use a hybrid model with WRF-ARV version 3.4.1 (Skamarock et al., 2008) NWP as the physics-based part and Gaussian process nonlinear autoregressive exogenous model (GP-NARX) as the statistical part to predict ground level winds at a certain location. Two separate GP-NARX models are formed for wind components. Inputs to the GP-NARX are called regressors and are passed to GP-NARX as a vector. The model then predicts the output value based on the inputs and on the training data.

The structure of GP-NARX can be presented in the form of a nonlinear stochastic recurrent equation like (Kocijan, 2016)

$$\begin{aligned} y(t) = & f\left(y(t-1), \dots, y(t-n),\right. \\ & \left.\boldsymbol{u}(t-1), \dots, \boldsymbol{u}(t-m)\right) + \nu. \end{aligned} \tag{1}$$

The vector $\boldsymbol{u}$ consists of the model inputs, while the output signal is named y. The discrete parameter t marks the time for which the prediction is made and its value increases by 1 in every time step. The term ν is noise. In the case of GP-NARX, the nonlinear function f is a GP.

The GP-NARX models are validated by simulation. The model is used to calculate its output variable throughout a time period. Measurements and NWP outputs are used as input regressor values for the outside signals. When the wind component that the model predicts is used as the model input, delayed model output is used, except at the beginning when measured wind component values before the simulation time period are used. The model output and the measured value time series can then be compared. 1-step ahead predictions or predictions with a larger fixed prediction horizon are not analysed in this study.

2.1 Regressor selection

Increasing the amount of training data used improves the model performance. However, there are practical limits on the training set size determined by the available historical data and available computing resources. Limited size of the training set in turn limits the number of regressors that can be used. Specifically, additional regressors, in particular irrelevant ones, increase the required amount of training data. For this reason we desire having a small number of regressors that are all relevant and that, taken as a whole, contain as much information relevant for predicting the output as possible.

We thus limit ourselves to a certain number of regressors that we select from all possible regressor candidates using a regressor selection method. The tool we use for regressor selection is ProOpter IVS (Gradišar et al., 2015). We test different criteria for regressor selection – LIP, MI, and PMI are part of ProOpter IVS and we have added RELIEFF (Kononenko et al., 1997) as implemented in MATLAB® built-in function `relieff()` to ProOpter IVS. We use the default values of the free parameters of the methods because optimising them would be excessive. However, of the four tested methods, we present the results with the one that gave best results, which is LIP.

ProOpter IVS works by ranking the regressor candidates. We use some of the best ones as regressors.

2.2 Gaussian process modelling

By definition, GP is a stochastic process $f(\boldsymbol{z})$ for which any finite set of values $\{f(\boldsymbol{z}_1), f(\boldsymbol{z}_2), \dots, f(\boldsymbol{z}_M)\}$ is jointly normally distributed. For a selection of points $\boldsymbol{z}_1, \dots, \boldsymbol{z}_M$, the joint probability density function of $f(\boldsymbol{z}_1), \dots, f(\boldsymbol{z}_M)$ is $p(f(\boldsymbol{z}_1), \dots, f(\boldsymbol{z}_M)) = \mathcal{N}(\boldsymbol{m}, \boldsymbol{\Sigma})$, where $\boldsymbol{m}$ is the mean vector and $\boldsymbol{\Sigma}$ is the covariance matrix.

A way to construct a GP is computing the elements m_i of $\boldsymbol{m}$ as values of a *mean function* $m(\boldsymbol{z})$, $m_i = m(\boldsymbol{z}_i)$. Similarly, the matrix elements $\boldsymbol{\Sigma}_{ij}$ can be values of a *kernel function* or *covariance function* $k(\boldsymbol{z}, \boldsymbol{z}')$, $\boldsymbol{\Sigma}_{ij} = k(\boldsymbol{z}_i, \boldsymbol{z}_j)$. The role of covariance function can be served by any function that results in a positive, semi-definite covariance matrix (Kocijan, 2016).

The goal of stochastic modelling of a dynamic system is to determine the relation between the input $\boldsymbol{z}$ and the output y. It can be given in the form $y(\boldsymbol{z}) = f(\boldsymbol{z}) + \nu$, where $f(\boldsymbol{z})$ is an underlying function and ν is the error, composed of both measurement and model errors. For a model to be complete, the error has to be modelled. Our choice of the error model is white noise $\nu \sim \mathcal{N}\left(0, \sigma_{\boldsymbol{z}}{}^2\right)$. In the case of GP modelling, the modelled measurement $y(\boldsymbol{z})$ is based on the sum of a GP and the noise signal. The choice of covariance function of the GP is based on our knowledge of the system, and as the *prior* estimate of the mean function, typically $m(\boldsymbol{z}_i) \equiv 0$ is used. When the *training data* is taken into account, the *posterior* distribution $p(y(\boldsymbol{z}))$ results. By training data we mean the regression matrix $\mathbf{Z}$ of the regression vectors, $\mathbf{Z} = [\boldsymbol{z}_1, \dots, \boldsymbol{z}_N]$, and the vector $\boldsymbol{y} = [y_1, \dots, y_N]^T$ of training outputs y_i. We label it as $\mathcal{D} = \{\mathbf{Z}, \boldsymbol{y}\}$. We assume the training outputs represent noisy realizations of the GP model, $f(\boldsymbol{z}_i) = y(\boldsymbol{z}_i) + \nu_i$, $p(\nu_1, \dots, \nu_N) = \mathcal{N}(0, \boldsymbol{\Sigma}_\nu)$. We assume the noise is identically and independently distributed with the covariance matrix $\boldsymbol{\Sigma}_\nu = \sigma_\nu{}^2\mathbf{I}$. We introduce matrix $\mathbf{K}$ defined as $\mathbf{K} = \boldsymbol{\Sigma} + \boldsymbol{\Sigma}_\nu$. At a test point $\boldsymbol{z}^*$, we predict the distribution of the process as (Kocijan, 2016)

$$p(f(\boldsymbol{z}^*) | \mathcal{D}, \boldsymbol{z}^*) = \mathcal{N}\left(\mu(\boldsymbol{z}^*), \sigma^2(\boldsymbol{z}^*)\right), \tag{2}$$

where $\mu(\boldsymbol{z}^*)$ and $\sigma^2(\boldsymbol{z}^*)$ are defined by

$$\mu(\boldsymbol{z}^*) = \boldsymbol{k}^{\mathrm{T}}\mathbf{K}^{-1}\boldsymbol{y} \tag{3}$$

$$\sigma^2(\boldsymbol{z}^*) = \kappa(\boldsymbol{z}^*) - \boldsymbol{k}^{\mathrm{T}}\mathbf{K}^{-1}\boldsymbol{k}. \tag{4}$$

Furthermore, $\boldsymbol{k}$ and κ are defined as $\boldsymbol{k}_{1\times N} = k(\boldsymbol{z}^*, \mathbf{Z})^{\mathrm{T}}$ and $\kappa_{1\times 1} = k(\boldsymbol{z}^*, \boldsymbol{z}^*)$. To obtain the probability density of the measured output y^* at $\boldsymbol{z}^*$, noise has to be taken into account. The resulting expression is $p(y^* | \mathcal{D}, \boldsymbol{z}^*) = \mathcal{N}\left(\mu(\boldsymbol{z}^*), \sigma^2(\boldsymbol{z}^*) + \sigma_\nu{}^2\right)$.

The knowledge of the system is typically not sufficient to fully define the noise variance and a suitable covariance function in advance. Instead we determine the parameters of the covariance function $\mathbf{\Theta}$ that are named *hyperparameters* from the training data with optimization – we maximize the likelihood $p\left(\mathbf{\Theta}|\mathbf{Z},\boldsymbol{y}\right)$. If we start from the assumption that every value of each hyperparameter is equally likely, we get (Kocijan, 2016)

$$p\left(\mathbf{\Theta}|\mathbf{Z},\boldsymbol{y}\right) \propto p\left(\boldsymbol{y}|\mathbf{Z},\mathbf{\Theta}\right). \tag{5}$$

Taking the logarithm of the normal distribution on the right, we get (Kocijan, 2016)

$$\log p\left(\boldsymbol{y}|\mathbf{Z},\mathbf{\Theta}\right) = -\frac{n}{2}\log 2\pi - \frac{1}{2}\log|\mathbf{K}| - \frac{1}{2}\boldsymbol{y}^{\mathrm{T}}\mathbf{K}^{-1}\boldsymbol{y} \tag{6}$$

and use that expression for optimization.

2.3 Simulation

The distribution of $y\left(t\right)$ from (1) is not simple because of the non-linearity of f. We thus do not simulate analytically and use the Monte Carlo method instead. To obtain a realization, we sample the values of ν and f for all values of t from their distributions. Both distributions are Gaussian, including the distribution of the values of f because the inputs of f are sampled values, meaning that their distributions are Dirac delta. We use a sample of realizations to estimate the expected value and variance of the modelled variable.

3. DESCRIPTION OF THE EXPERIMENTS

3.1 The studied example

We are predicting west-east and south-north components of the wind at ground level (10 m above ground) at Stolp meteorological station that is adjacent to Krško NPP. In the surrounding area of 25 km × 25 km, data from 5 other meteorological stations (Brežice, Cerklje, Cerklje Airport, Krško, Lisca) is available to us. They measure wind, temperature, relative humidity, and air pressure, not all of them at every station. At Stolp weather station, temperature is measured on 4 levels up to 70 m above ground and relative humidity on 3 levels up to 40 m high using sensors on a tower. In total, there are 32 signals from meteorological stations available. There is also SODAR (Brown and Hall Jr., 1978) next to the tower, measuring all 3 wind components in 24 layers. We only use the lowermost 5 layers as they are the most reliable ones, providing 15 signals. We use predictions for 1 cell of the NWP model, which is 7 signals, and one more signal (diffuse solar irradiation) is generated from NWP signals using an artificial neural network and treated like a NWP signal. We also produce 4 mathematical signals representing time of day and season, they are sine and cosine signals with period of 1 day and 1 year starting at the beginning of the day or the year. There are 59 signals in total, data is available for 6 years starting at the beginning of 2012, and one sample is available every 30 minutes.

From the signals, we generate 122 possible regressors: we use two delays for each measurement corresponding to $t-1$ and $t-2$ where t is the time index corresponding to the time for which the prediction is made. For the WRF signals, we are not limited to the past and we use 3 values corresponding to $t-1$, t, and $t+1$, while for the mathematical signals, we deem the value at t sufficient.

Table 1. Best regressors according to ProOpter IVS LIP method. They are listed from best one on. A positive number in the "delay" field means that the signal value that is used corresponds to a time before the time to which the prediction corresponds, and vice versa.

Best regressors for W-E wind

source	variable	delay
Stolp	W-E wind	1
WRF	W-E wind	−1
Cerklje Airport	W-E wind	1
Krško	W-E wind	1
Cerklje Airport	W-E wind	2
Brežice	W-E wind	1
Krško	air temperature	1
WRF	global solar irradiation	−1
Cerklje Airport	S-N wind	1
WRF	global solar irradiation	1
SODAR	W-E wind, layer 1	1
Cerklje Airport	air temperature	1
Cerklje	air pressure	2
Stolp	air pressure	1
Krško	S-N wind	1

Best regressors for S-N wind

source	variable	delay
Stolp	S-N wind	1
Brežice	S-N wind	1
Krško	S-N wind	1
Cerklje Airport	W-E wind	1
Cerklje Airport	S-N wind	1
Cerklje Airport	S-N wind	2
WRF	S-N wind	−1
SODAR	S-N wind, layer 3	1
WRF	global solar irradiation	−1
Stolp	air temperature, 70 m	2
Lisca	S-N wind	1
Krško	air temperature	1
WRF	W-E wind	−1
Cerklje Airport	air pressure	1
Cerklje Airport	air temperature	2

We rank the regressors using ProOpter IVS and use at most 15 best of them. The 15 best regressors for predicting each wind component are shown in table 1.

3.2 Model construction

For the covariance function, we pick squared exponential function. Its form, expressed with components, is

$$k\left(\boldsymbol{z},\boldsymbol{z}'\right) = {\sigma_f}^2 \exp\left[-\frac{1}{2}\sum_{d=1}^{D}\frac{\left(z_d - {z'}_d\right)^2}{{l_d}^2}\right]. \tag{7}$$

The $D+1$ hyperparameters are σ_f and l_d for all the values of d. The noise covariance σ_ν also has to be determined. We replace σ_ν with another parameter $\lambda = \frac{{\sigma_\nu}^2}{{\sigma_f}^2}$. The optimum value of σ_f can then be computed using the relationship (Stepančič and Kocijan, 2017)

$$\sigma_f = \left(\frac{N}{\boldsymbol{y}^{\mathrm{T}}(\mathbf{\Sigma}+\lambda\mathbf{I})^{-1}\boldsymbol{y}}\right)^{\frac{1}{2}}. \tag{8}$$

We use conjugate gradient method (Rasmussen and Williams, 2006; Rasmussen and Nickisch, 2010) to determine the optimal values of λ and of the remaining hyperparameters. The criterion function is given in (6).

3.3 Reduction of training data

Too much training data is available and not all of it can be used due to computing limitations. We thus select a subset of it as our training data set. The obvious way of doing it is downsampling. We select every s-th sample from the training data in order to change the training data set's size by the factor $1/s$ (where s is a number bigger than 1). This selection method is essentially random – whether an available data point ends up in the training data set or not is determined by chance.

One can use a more advanced and hopefully smarter method to select the data points to be used for training with the intention of achieving better modelling results than with random selection. Several data selection methods have been presented in the literature (Khosravani et al., 2016, 2017). The smart selection method we use is based on Euclidean distance between training points. Every data point is treated as a vector with (normalized) regressor and measured output values as its coordinates. For every point in the initial training data, the distance to its nearest neighbour is computed. The points whose distances to their nearest neighbours are large are kept in the output training set, the ones with neighbours nearby are rejected. The procedure is done iteratively: in each step, 95 % of the points are kept and 5 % are discarded, and the obtained output training set is reduced again until the desired number of training points is reached.

Statistical models work well in the regions of regression space where the density of training points is high and perform badly when they extrapolate (Kocijan, 2016). By making sure the training points are spread out and not randomly clustered, the model can be expected to perform better on average. We check this hypothesis by comparing the model performance with smartly selected training points to the performance of a model with the points randomly selected.

3.4 Model evaluation

Two simulations are performed for each model, one for the first 14 days of June and one for the first 14 days of December 2017. Two time periods are chosen in order not to focus on a single season, and data for 2017 is used because newer data is more complete (since missing data would make simulation impossible, every missing value is filled in with the value obtained with linear interpolation between the first preceding and first succeeding valid value). Two performance measures, normalized root-mean-square error (NRMSE) and mean standardised log loss (MSLL), of the simulation result are then computed.

NRMSE is expressed as

$$\text{NRMSE} = 1 - \frac{\|\boldsymbol{y} - \boldsymbol{\mu}\|}{\|\boldsymbol{y} - E\left(\boldsymbol{y}\right)\|}, \qquad (9)$$

where $\boldsymbol{y}$ are measured values and $\boldsymbol{\mu}$ are mean predicted values. Bigger NRMSE value is better, perfect model has NRMSE $= 1$ and there is no lower limit. NRMSE is easily generalised to be used in cases when the model predicts a vector quantity such as wind in two dimensions.

MSLL is computed as (Rasmussen and Williams, 2006)

$$\begin{aligned}\text{MSLL} =& \frac{1}{2N}\sum\left(\ln\left(\sigma_i^2\right) + \frac{\left(\mu_i - y_i\right)^2}{\sigma_i^2}\right)\\ &- \frac{1}{2N}\sum\left(\ln\left(\sigma_y^2\right) + \frac{\left(y_i - E(\boldsymbol{y})\right)^2}{\sigma_y^2}\right).\end{aligned} \qquad (10)$$

The symbol $E(\boldsymbol{y})$ stands for the mean of the measured values, σ_y is variance of the measured values, and σ_i is prediction variance. MSLL is supposed to be around 0 for simple methods and negative for better ones (Rasmussen and Williams, 2006). Its advantage is that it takes prediction variance into account.

4. RESULTS

4.1 Basic models

The training data that GP-NARX requires is obtained from the signals for the years 2012–2017 without the parts of June and December 2017 that are kept for validation. Only data points with all the 15 regressors and the output signal available are used. However, the number of available complete data points is too big for further computation, so the random reduction method from section 3.3 is utilized. Only every 19-th one is used as a part of the training data set. 19 is used as a quotient because it is a coprime of 48, the number of data points in a day, minimizing the probability of biases in the sample. Since regressors for the two models are different, the final number of data points used in training is different between the two as well: there are 2545 for the W-E wind model and 2114 for the S-N model.

For each model, both simulations are then performed and evaluated as a whole and the performance measures are computed. The W-E model achieves MSLL of −0.585, and MSLL of the S-N model is −0.512. NRMSE is calculated for wind as a 2D vector resulting from simulations of both models and equals 0.409.

Simulation result for the first week of December for W-E model is shown in Fig. 1 as an example.

4.2 Models with smart training set selection

The training data is obtained from the signals for 2012–2017 except the parts that are kept for validation through simulation. Only data points with complete regression vectors are used. With the smart method from section 3.3, the number of training points to be used is reduced to 2545 for the W-E wind model and 2114 for the S-N model, the same training point numbers as in section 4.1.

The simulations are performed for each model and evaluated as a whole. The W-E model achieves MSLL of −0.690, and MSLL of the S-N model is −0.261. NRMSE is calculated for wind as a 2D vector and equals 0.440.

Simulation result for the first week of December for W-E model is shown in Fig. 2.

4.3 Models with fewer regressors

The same operations as in sections 4.1 and 4.2 are performed for models with every number of regressors between

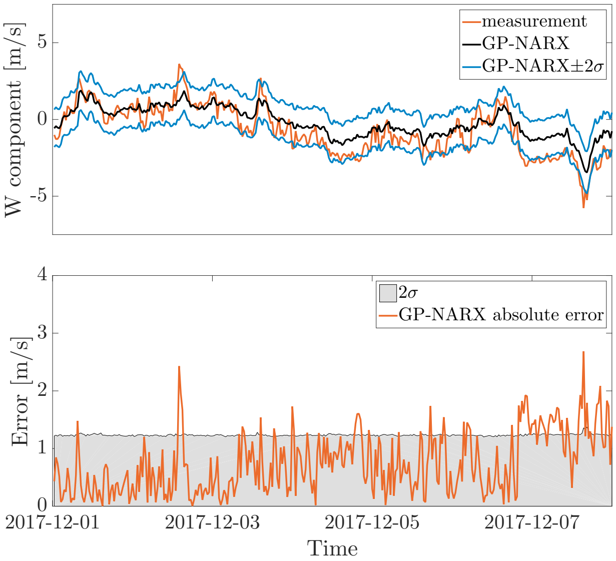

Fig. 1. Measured and modelled W-E wind speed in the first week of December 2017 for the model with 15 regressors and random selection of training points. The upper panel shows the measured value, the mean value of the simulation result, and the band around the mean value with half-width of double standard deviation. In the lower panel, the error (difference between the measurement and the modelled mean value) and the standard deviation are compared.

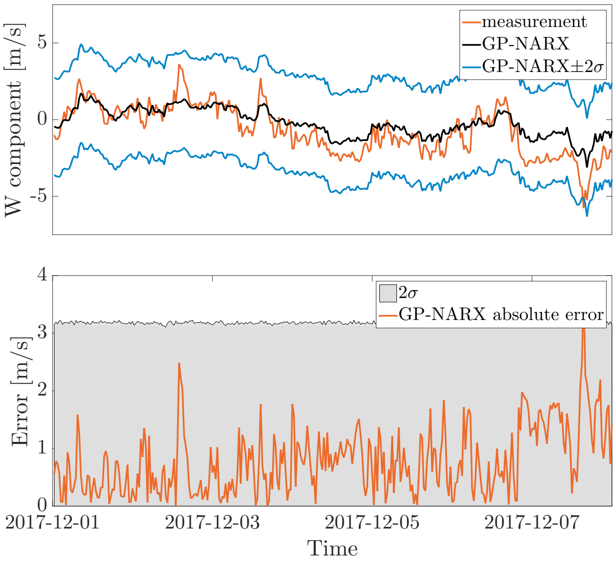

Fig. 2. Measured and modelled W-E wind speed in the first week of December 2017 for the model with 15 regressors and smart selection of training points. The upper panel shows the measured value, the mean value of the simulation result, and the band around the mean value with half-width of double standard deviation. In the lower panel, the error (difference between the measurement and the modelled mean value) and the standard deviation are compared.

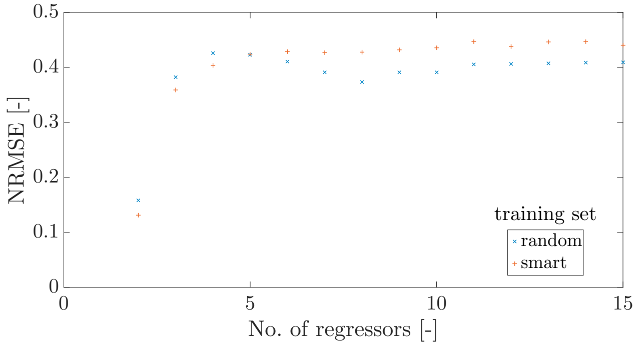

Fig. 3. NRMSE value for the simulation results as a function of the number of regressors used. Results for both models resulting from randomly and smartly selected training points are shown.

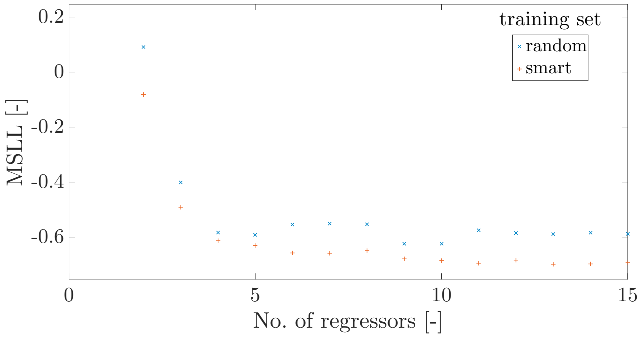

Fig. 4. MSLL value for the simulation results of the west-east wind component model as a function of the number of regressors used. Results for both models resulting from randomly and smartly selected training points are shown.

2 and 15. In every case, the best regressors (the ones located highest in the relevant part of the table 1) are used. All the W-E models use 2545 training points and all the S-N ones 2114 training points. To get these numbers of points with the random selection method, it has to be modified as the number of available complete training points increases with decreasing number of regressors. The points are downsampled with the highest prime number that leaves enough points and the extra points at the end of the dataset are discarded.

The values of the performance measures are shown graphically, NRMSE in Fig. 3 and MSLL in Figs. 4 and 5.

5. DISCUSSION

The distance method is meant to select the training points in such a way as to uniformly cover the regression space, or at least the part of it that the process / training data covers. In contrast, selecting the points randomly causes the chosen point distribution to reflect the training point distribution. The distance method thus better preserves information about the more sparsely sampled parts of the regression space, preventing extrapolation and improving model behaviour in the less common situations.

However, the regression space is too big (too many regressors bring too many dimensions) to be uniformly covered with a reasonable number of training points. The distance

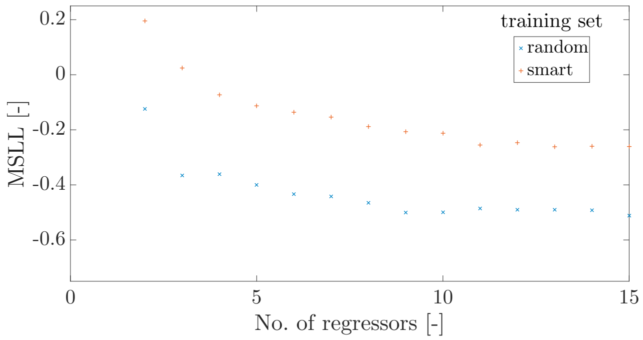

Fig. 5. MSLL value for the simulation results of the south-north wind component model as a function of the number of regressors used.

method results in the points being chosen more on the outskirts of the process than in the central crowded part. In the example of south–north wind component, the resulting reduction in extrapolation appears to be outweighed by less dense training points in the most frequented parts of the space. In the case of the west–east wind direction, the intended effects of the distance method happen to outweigh the side effects, leading to an improvement of the model.

In the case of 15 regressors, the number of training points cannot be increased enough to cover the regression space uniformly because the space is too big. Another option is decreasing the number of regressors to shrink the space and observing the performance of the smart selection method in that case. We reduce the number of regressors while keeping the number of training points constant. As shown in Figs. 3–5, the smart selection method does not outperform random selection in the case of fewer regressors. Possibly the models with fewer regressors are lacking necessary information for modelling the output and points that are close to each other in the regressor subspace do not necessarily correspond to similar system states.

6. CONCLUSION

In development of a GP-NARX model as part of a hybrid model for predicting local meteorological variables, we encounter the issue of being able to use only a small part of the available training data because of computational limitations. A trivial solution of this problem would be to use a randomly selected subset of the data for training.

We seek a different way of selecting the subset of the data to use that would lead to better results than random selection. The method we test, which is supported by mathematical reasoning, fails to provide a consistent and significant improvement. Further exploration indicates that for the given system and the given allowed number of training data points, random selection results in a training set that is close to optimal. It seems that the system is too complex to be described with so few training data points regardless of the exact selection. Alternatively, some of the model dynamics is missing, there may be influences on the output variable that cannot be recognised from the available regressors, in which case neither better choice nor bigger number of training data points would help.

REFERENCES

Božnar, M.Z., Mlakar, P., and Grašič, B. (2012). Short-term fine resolution WRF forecast data validation in complex terrain in Slovenia. *International Journal of Environment and Pollution*, 50(1-4), 12–21. doi:10.1504/IJEP.2012.051176.

Brown, E.H. and Hall Jr., F.F. (1978). Advances in atmospheric acoustics. *Reviews of Geophysics*, 16(1), 47–110. doi:10.1029/RG016i001p00047.

Gradišar, D., Glavan, M., Strmčnik, S., and Mušič, G. (2015). ProOpter: An advanced platform for production analysis and optimization. *Computers in Industry*, 70, 102 – 115. doi:10.1016/j.compind.2015.02.010.

Khosravani, H., Ruano, A., and Ferreira, P. (2017). A comparison of four data selection methods for artificial neural networks and support vector machines. *IFAC-PapersOnLine*, 50(1), 11227 – 11232. doi:10.1016/j.ifacol.2017.08.1577.

Khosravani, H., Ruano, A., and Ferreira, P. (2016). A convex hull-based data selection method for data driven models. *Applied Soft Computing*, 47, 515 – 533. doi:10.1016/j.asoc.2016.06.014.

Kocijan, J. (2016). *Modelling and Control of Dynamic Systems Using Gaussian Process Models*. Advances in Industrial Control. Springer International Publishing. doi:10.1007/978-3-319-21021-6.

Kocijan, J., Girard, A., Banko, B., and Murray-Smith, R. (2005). Dynamic systems identification with Gaussian processes. *Mathematical and Computer Modelling of Dynamical Systems*, 11(4), 411–424. doi:10.1080/13873950500068567.

Kononenko, I., Šimec, E., and Robnik-Šikonja, M. (1997). Overcoming the myopia of inductive learning algorithms with RELIEFF. *Applied Intelligence*, 7(1), 39–55. doi:10.1023/A:1008280620621.

Petelin, D., Grancharova, A., and Kocijan, J. (2013). Evolving Gaussian process models for prediction of ozone concentration in the air. *Simulation Modelling Practice and Theory*, 33, 68–80.

Petelin, D., Mlakar, P., Božnar, M.Z., Grašič, B., and Kocijan, J. (2015). Ozone forecasting using an online updating Gaussian-process model. *International Journal of Environment and Pollution*, 57(3-4), 111–122.

Potter, C.W. and Negnevitsky, M. (2006). Very short-term wind forecasting for Tasmanian power generation. *IEEE Transactions on Power Systems*, 21(2), 965–972. doi:10.1109/TPWRS.2006.873421.

Rasmussen, C.E. and Nickisch, H. (2010). GAUSSIAN PROCESS REGRESSION AND CLASSIFICATION Toolbox version 3.1 for GNU Octave 3.2.x and Matlab 7.x.

Rasmussen, C.E. and Williams, C.K.I. (2006). Gaussian processes for machine learning. MIT Press.

Skamarock, W.C., Klemp, J.B., Dudhia, J., Gill, D.O., Barker, D.M., Wang, W., and Powers, J.G. (2008). A description of the Advanced Research WRF version 3. *NCAR Tech. Note NCAR/TN-475+STR*.

Stepančič, M. and Kocijan, J. (2017). On-line identification with regularised evolving Gaussian process. In *2017 Evolving and Adaptive Intelligent Systems (EAIS)*, 1–7. doi:10.1109/EAIS.2017.7954820.

Wiener, N. (1961). *Cybernetics Or Control and Communication in the Animal and the Machine*. M.I.T. Press.