Atmos. Meas. Tech., 7, 4387–4399, 2014
www.atmos-meas-tech.net/7/4387/2014/
doi:10.5194/amt-7-4387-2014

Atmospheric Measurement Techniques
Open Access

Regression models tolerant to massively missing data: a case study in solar-radiation nowcasting

I. Žliobaitė[1,2], J. Hollmén[1,2], and H. Junninen[3]

[1]Aalto University, Department of Information and Computer Science, Espoo, Finland
[2]Helsinki Institute for Information Technology (HIIT), Helsinki, Finland
[3]Department of Physics, University of Helsinki, Helsinki, Finland

Correspondence to: I. Žliobaitė (indre.zliobaite@aalto.fi)

Received: 14 April 2014 – Published in Atmos. Meas. Tech. Discuss.: 16 July 2014
Revised: 6 November 2014 – Accepted: 14 November 2014 – Published: 11 December 2014

Abstract. Statistical models for environmental monitoring strongly rely on automatic data acquisition systems that use various physical sensors. Often, sensor readings are missing for extended periods of time, while model outputs need to be continuously available in real time. With a case study in solar-radiation nowcasting, we investigate how to deal with massively missing data (around 50 % of the time some data are unavailable) in such situations. Our goal is to analyze characteristics of missing data and recommend a strategy for deploying regression models which would be robust to missing data in situations where data are massively missing. We are after one model that performs well at all times, with and without data gaps. Due to the need to provide instantaneous outputs with minimum energy consumption for computing in the data streaming setting, we dismiss computationally demanding data imputation methods and resort to a mean replacement, accompanied with a robust regression model. We use an established strategy for assessing different regression models and for determining how many missing sensor readings can be tolerated before model outputs become obsolete. We experimentally analyze the accuracies and robustness to missing data of seven linear regression models. We recommend using the regularized PCA regression with our established guideline in training regression models, which themselves are robust to missing data.

1 Introduction

Environmental monitoring strongly relies on automatic data acquisition systems, using various physical sensors. For instance, stations measuring atmosphere ecosystem relationships (SMEAR) stations[1] measure the relationship of atmosphere and forest in the boreal climate zone (Hari and Kulmala, 2005). The stations are equipped with an extensive range of measurement instruments: atmospheric and flux measurements, irradiation and flux measurements, tree physiology measurements, soil and soil-water measurements, and solar irradiance. Due to the continuous flux of measurements, the setup can be analyzed in the context of streaming data (Babcock et al., 2002; Aggarwal, 2007). Streaming-data analysis is different from the traditional retrospective data analysis, where data are first collected, cleaned, preprocessed, and then analyzed. Streaming data arrive continuously and need to be analyzed in real time. Statistical models built on such streaming data (see, e.g., Lu et al., 2006, Hrust et al., 2009, and Menut and Bessagnet, 2010) need to operate continuously and provide outputs in real time.

Physical sensors are exposed to various risks due to severe environmental conditions, exposure to physical damage, or battery drainage. Under such circumstances it is very common to encounter time intervals when readings from some of the sensors are missing from the database. A lot of advanced missing-value imputation schemes have been developed (Junninen et al., 2004; Allison, 2001), primarily targeting offline exploratory data analysis, where computational resources are practically unlimited, while it is critical to re-

[1]http://www.atm.helsinki.fi/SMEAR/

construct data as accurately as possible. A simple mean replacement remains popular in regression modeling (Kadlec et al., 2009) in situations where real-time outputs are needed and computational resources and time are limited, but the input data does not need to be reconstructed perfectly accurately, as long as model outputs remain correct.

The goal of this study is to experimentally analyze the performance and robustness of linear regression models with regard to massively missing data for operation in resource-aware settings. We consider situations where data are massively missing, which means that around 50 % of the time at least one sensor does not deliver readings and there is no single sensor that dominates the missing data; data from any sensor can be missing. In such a situation, readings from input sensors may be missing for extended periods of time; nevertheless, model outputs need to be produced continuously and delivered in real time; not producing model outputs when some data are missing is not an option. We aim at building one regression model that is robust in performance; i.e., the expected performance is stable, no matter how many sensor readings are missing.

We present a case study in solar-radiation nowcasting using meteorological sensor data as inputs, where multiple sensor failures happen frequently due to environmental and operational reasons. We analyze the performance of seven linear regression models coupled with the mean replacement of missing values and provide recommendations for robust and accurate modeling in such circumstances. Nowcasting refers to predicting the *current* values from other measurements and is different from forecasting, which aims at predict future values from the past values.

The paper presents a case study in which our earlier published results (Žliobaitė and Hollmén, 2013) are put into practice for solving a solar-radiation nowcasting task in the context of a SMEAR measurement station (Hari and Kulmala, 2005). A reader interested in the theoretical underpinnings of our approach and a follow-up is advised to refer to studies by Žliobaitė and Hollmén (2013, 2014); the current paper focuses on practical implications of the results and demonstrates how regression problems with lots of missing data can be successfully solved with our recommended scheme. The results apply to the case of linear regression coupled with the mean replacement of missing values. We assume that the uncertainty of the sensor measurements is stable over time when the measurements are available.

Research attention to solar-radiation nowcasting and short-term forecasting using statistical-data-driven models is increasing due to the growing popularity of solar-energy power plants that need solar-radiation estimates for planning. Research studies mostly focus on searching for a suitable statistical modeling technique: artificial neural networks (Marquez and Coimbra, 2011), autoregressive time series models (Bacher et al., 2009), Markov models (Bhardwaj et al., 2013), or optimally integrating different data sources, such as meteorological variables, ground and remote sensing observations, or satellite images (Hammer et al., 1999; Vuilleumier et al., 2011). We are not aware of any research work addressing the problem of massively missing values in solar-radiation nowcasting.

The rest of the paper is organized as follows. Section 2 describes the SMEAR data used in the case study, the methodology of the modeling, and the experimental protocol. Section 3 presents and discusses the results of the case study. Section 4 summarizes the contributions and concludes the study.

2 Materials and methods

2.1 Data

We use a data stream recorded at SMEAR II station in Hyytiälä, Finland (Junninen et al., 2009) (61°50′51″ N, 24°17′41″ E; 181 m a.s.l.), measuring relationships between the forest ecosystem and atmosphere. We use data covering a period of 7 years (April 2005–April 2013), recorded at every 30 min from 37 observation sensors. The raw data coming from the station have on average 7 % of missing values. Missing values may occur due to the occasional failure of measuring sensors, wear and tear, or variations in electricity power supply. Some data are missing up to 50 % of the time. There is no single sensor that would provide non-interrupted readings over those 5 years; for any sensor from 1 % (about 4 days per year) up to 25 % (3 months per year) values are missing.

The task is to nowcast the current level of solar radiation from the meteorological sensor data, given in Table 1. The incoming radiation to Earth is constant with the accuracy we require at a given day and hour of the year. The only unknown is the absorption to the atmosphere and, more importantly, to the clouds and anthropogenic pollution plumes. Hence, an interesting variable to infer is the cloudiness, or, in other words, the deviation of the measured radiation from the theoretical maximum. In this schema other meteorological parameters could be used to estimate the cloudiness, and this can further be used to calculate the actual radiation, but the primary variable to nowcast is the difference between the theoretical and actual radiation.

This nowcasting task would be relevant to the stations where no radiation measures are available. The station SMEAR II, from which the input data originate, is able to measure solar radiation; hence, the true values are present for us for evaluation purposes. However, instrumentation for measuring solar radiation is not always available. Small meteorological observation stations may not be able to have solar radiation measured, but it may be interesting to nowcast radiation from meteorological data that is available anyway.

In this study our target variable is defined as the ratio of the actual radiation to the theoretical maximum radiation. This gives a value between 0 and 100 %, where 100 % indicates

Table 1. Sensors for the case study: SWS – surface wetness sensor; P – pressure; T – temperature; WS – wind speed; WD – wind direction; RH – relative humidity; RH Td – relative humidity calculated using dew point; PTG – potential temperature gradient; Vis – visibility.

Index	measurement	height	missing values
1	Rain	18.0 m	1 %
2	SWS	18.0 m	1 %
3	Dew point	18.0 m	18 %
4	P	0.0 m	2 %
5	T	4.2 m	16 %
6	T	8.4 m	3 %
7	T	16.8 m	2 %
8	T	33.6 m	2 %
9	T	50.4 m	2 %
10	T	67.2 m	2 %
11	WS	33.6 m	9 %
12	WS	8.4 m	5 %
13	WS	16.8 m	3 %
14	WS	33.6 m	9 %
15	WS	74.0 m	25 %
16	WD avr		2 %
17	WD ultrasonic	8.4 m	7 %
18	WD ultrasonic	16.8 m	4 %
19	WD ultrasonic	33.6 m	9 %
20	WD ultrasonic	74.0 m	23 %
21	RH	4.2 m	21 %
22	RH	8.4 m	9 %
23	RH	16.8 m	7 %
24	RH	33.6 m	7 %
25	RH	50.4 m	9 %
26	RH	67.2 m	6 %
27	RH Td	18.0 m	20 %
28	PTG		5 %
29	Visibility	18.0 m	1 %
30	Vis-min	18.0 m	1 %
31	Vis-max	18.0 m	1 %
32	Precipitation intensity	18.0 m	1 %
33	Preci-min	18.0 m	1 %
34	Preci-max	18.0 m	1 %
35	Precipitation	18.0 m	1 %
36	Snowfall	18.0 m	1 %
37	Global RADIATION	18.0 m	1 %

that all the theoretically possible radiation is actually incoming. The sensor Global RADIATION (Table 1) is not used as an input into the nowcasting model; it is only used for evaluating the nowcasting accuracy. It indicates the actual radiation and is used in forming the target variable.

The theoretical maximum radiation is calculated using MIDC (Measurement and Instrumentation Data Center) SOLPOS (Solar Position and Intensity) Calculator[2]. SOLPOS is a computational tool that calculates the apparent solar position and intensity (theoretical maximum solar energy)

[2]http://www.nrel.gov/midc/solpos/solpos.html

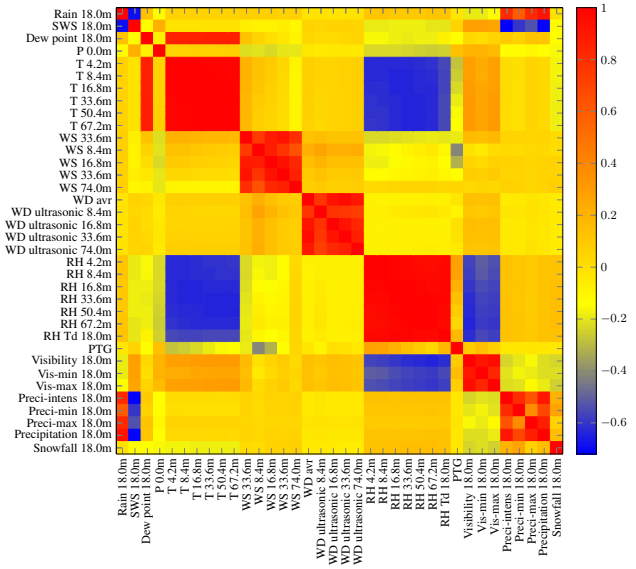

Figure 1. Correlations between input sensors.

based on the date, time, and location on Earth. The tool is developed and maintained by The National Renewable Energy Laboratory, which is operated for the US Department of Energy by the Alliance for Sustainable Energy. The calculations are based on established models for solar position, reported in Michalsky (1988) and other sources.

The following input parameters were used: lat – 61.8475; long – 24.29472; time zone – 2 (location parameters); surface pressure – 990 mbar; ambient dry-bulb temperature – 3 °C; azimuth of panel surface – 180°; degrees of tilt from horizontal of panel – 0; solar irradiance constant – 1360.8 W m^{-2} (Kopp and Lean, 2011); shadow-band width – 7.6 cm; shadow-band radius – 31.7 cm; shadow-band sky factor – 0.04; interval of a measurement period – 0 s.

Often sensor readings are correlated with each other. Figure 1 visualizes the pairwise correlations computed over non-missing data. We see distinct blocks of positive and negative correlations. For instance, relative humidity (RH) is negatively correlated with temperature (T).

2.2 Prerequisites

2.2.1 Setting

Suppose we have r sources generating streaming data (e.g., weather observation sensors). Data are recorded in multidimensional vectors $\boldsymbol{x} \in \mathbb{R}^r$. Our task is to nowcast the target variable $y \in \mathbb{R}^1$ (e.g., solar radiation) using these sensor readings as inputs. The regression model is then $y = f(\boldsymbol{x}) = f(x_1, \ldots, x_r)$, and the corresponding learning task is to approximate function f from the available input–output data. It is important to note that we do not make use of temporal information of the variables; that is, we predict the value of

the output y at time t, with the sensor readings available at the same time point t; hence, the task is referred to as nowcasting. With the time index in place, the regression model is $y^{(t)} = f(x_1{}^{(t)}, \ldots, x_r{}^{(t)})$. In the rest of the paper, we omit the time index t. For the identities of the sensors used in the case study ($r = 36$), see Table 1.

Data arrive in real time, and nowcasting outputs need to be delivered as soon as possible, in nearly real time. The nowcasting performance should be stable in the sense that the expected loss in accuracy due to possible missing values should be minimal. Bearing in mind that often environment monitoring sensors operate on batteries or autonomous power sources, the computational resources consumed for data processing, including missing-value imputation, should be minimal. We are after one model that performs well at all times, with and without data gaps.

2.2.2 Imputation of missing data

We assume that, when a sensor fails, missing values are automatically replaced with the *mean* values, which remains a popular approach in practice due to its simplicity and low user cost (Black et al., 2007; Kadlec et al., 2009; Enders, 2010). To keep the focus of the paper on the regression models tolerant to massively missing data, we also assume that there is no need to implement any driven missing-value detectors; the system knows when a value is missing.

In this study, we do not explore alternative imputation methods due to two reasons. Firstly, our main goal is to investigate the robustness of regression models to missing data rather than to select the best imputation scheme. Secondly, advanced model-based imputation methods such as linear interpolation, nearest neighbor imputation, and self-organizing map or multilayer perceptron methods (Junninen et al., 2004) typically are more accurate when the amount of missing data is small, but they lose their advantage when long missing-data gaps are expected. While multiple imputation methods (Junninen et al., 2004) bear relatively high computational costs and are favorable in one-off imputation operations, they are not very suitable for continuous online operations and imputation in real time. More importantly, such methods implicitly or explicitly assume that data are missing at random; i.e., a sensor value missing is independent both of observable variables and of unobservable parameters of interest. In reality this assumption ay often be violated, for instance by the sensors switching themselves off at low temperatures.

Bayesian approaches (Lerner et al., 2002; Ramoni and Sebastiani, 2001) present an interesting alternative for learning from incomplete data, but the goals and the task are somewhat different from what we are solving. In our setting, training data are abundant, and an initial model can be built from a subset that has no missing values. Bayesian nets can inherently learn from data with missing values, but once a model is ready, it does not seem to have any special mechanism for making predictions from incomplete data. In this case a Bayesian net would require an extra missing-value imputation approach, just like a linear regression.

One could create imputation models using knowledge about physical relationships between variables. However, when a lot of data is missing, such an approach would encounter a combinatorial explosion. One model would need to be available per each combination of missing variables, which requires building and maintaining 2^r models, where r is the number of input features.

2.2.3 Performance indicators

We use a nowcasting error as the main measure of performance, which is computed on a subset of data that was not used for parameter estimation (Hastie et al., 2001). The mean squared error (MSE) is a popular measure to quantify the discrepancy between the true target value y and the value output by the model, $\hat{y}$. MSE punishes large deviations from the true values; this is relevant for the environmental monitoring applications, where large errors are to be avoided. For practical interpretability, RMSE is often used, which is the square root of MSE. RMSE reports the error in the same units as the target variable. For a test data set of size n, MSE and RMSE are computed as

$$\mathrm{MSE} = \frac{1}{n}\sum_{l=1}^{n}(\hat{y}^{(l)} - y^{(l)})^2, \mathrm{RMSE} = \sqrt{\mathrm{MSE}}, \quad (1)$$

where $y^{(l)}$ is the true target value of the lth sample and $\hat{y}^{(l)}$ is the corresponding model output. In the experiments we report RMSE, which can be interpreted as an average deviation of model outputs from the true target values.

2.3 Computational methods

Linear regression model

For nowcasting we adopt linear regression models, which assume that the relationship between r input variables $\boldsymbol{x} = (x_1, \ldots, x_r)$ and the target variable y is linear. Without loss of generality we assume that the input data are standardized before modeling to have zero mean and unit standard deviation[3]. The regression model takes the form

$$y = b_1 x_1 + b_2 x_2 + \ldots + b_r x_r + \epsilon = \boldsymbol{x}\boldsymbol{\beta} + \epsilon, \quad (2)$$

where ϵ is the error variable and the vector $\boldsymbol{\beta} = (b_1, b_2, \ldots, b_r)^T$ contains the parameters of the linear model (regression coefficients). Since the data are assumed to have been standardized, there is no bias term in the model. In matrix form, the model is $\boldsymbol{y} = \mathbf{X}\boldsymbol{\beta} + \epsilon$, where $\mathbf{X}_{n\times r}$ is a sample

[3]For standardization we need to estimate the data mean m and the standard deviation s from a sample data set; then $x_{\mathrm{standardized}} = (x - m)/s$. For every variable we need to store the values m and s and apply the same procedure to all new incoming data before nowcasting.

data matrix containing n records from r sensors and $\boldsymbol{y}_{n\times 1}$ is a vector of the corresponding n target values.

2.4 Ordinary least squares

There are different ways to estimate the regression parameters (Hastie et al., 2001). Ordinary least squares (OLS) is a simple and probably the most common estimator. It minimizes the sum of squared residuals giving the following solution:

$$\hat{\boldsymbol{\beta}}_{\mathrm{OLS}} = \arg\min_{\boldsymbol{\beta}} \left((\boldsymbol{y} - \mathbf{X}\boldsymbol{\beta})^T (\boldsymbol{y} - \mathbf{X}\boldsymbol{\beta}) \right) = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\boldsymbol{y}. \tag{3}$$

Having estimated a regression model $\hat{\boldsymbol{\beta}}$, nowcasting on new data $\boldsymbol{x}_{\mathrm{new}}$ can be made as

$$\hat{y} = \boldsymbol{x}_{\mathrm{new}}\hat{\boldsymbol{\beta}}. \tag{4}$$

2.4.1 Regularization

If the input variables are correlated with each other, the optimization problem could result in poor estimates for the parameters. In such situations, regularization is often used for estimating the regression parameters. The Ridge regression (RR) (Hoerl and Kennard, 1970; Hastie et al., 2001) regularizes the regression coefficients by imposing a penalty on their magnitude. RR solution minimizes the cost function

$$\begin{aligned}\hat{\boldsymbol{\beta}}_{\mathrm{RR}} &= \arg\min_{\boldsymbol{\beta}} \left((\boldsymbol{y} - \mathbf{X}\boldsymbol{\beta})^T (\boldsymbol{y} - \mathbf{X}\boldsymbol{\beta}) + \lambda \boldsymbol{\beta}^T \boldsymbol{\beta} \right) \\ &= (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\boldsymbol{y},\end{aligned}$$

where $\lambda > 0$ controls the amount of shrinkage: the larger the value of λ, the greater the amount of shrinkage. $\mathbf{X}$ denotes the $n \times r$ training data set, and $\boldsymbol{y}$ is the $n \times 1$ vector of the true target values; $\mathbf{I}$ is the $r \times r$ identity matrix. Nowcasting outputs on new data $\boldsymbol{x}_{\mathrm{new}}$ can be produced as

$$\hat{y} = \boldsymbol{x}_{\mathrm{new}}\hat{\boldsymbol{\beta}}_{\mathrm{RR}}. \tag{5}$$

2.4.2 Principal component regression

Principal component (PCA) regression (Jolliffe, 2002) first transforms the input data by rotating them towards their principal components and then estimates the regression coefficients on the transformed data.

Let $\mathbf{X}_{n\times r}$ be the training data matrix, and $\mathbf{R}_{r\times k}$ is the matrix of k principal components, corresponding to the largest eigenvalues. Here, k is a user-defined parameter such that $1 \le k \le r$; if $k = r$, then PCA regression becomes the ordinary regression. Then OLS gives the following solution on the transformed input data:

$$\hat{\boldsymbol{\beta}}^*_{\mathrm{PCA}} = \arg\min_{\boldsymbol{\beta}} \left((\boldsymbol{y} - \mathbf{X}\mathbf{R}\boldsymbol{\beta}^*)^T (\boldsymbol{y} - \mathbf{X}\mathbf{R}\boldsymbol{\beta}^*) \right), \tag{6}$$

and in the original data space the solution is $\hat{\boldsymbol{\beta}}_{\mathrm{PCA}} = \mathbf{R}\hat{\boldsymbol{\beta}}^*_{\mathrm{PCA}}$. Nowcasting on new data $\boldsymbol{x}_{\mathrm{new}}$ can be made as

$$\hat{y} = \boldsymbol{x}_{\mathrm{new}}\mathbf{R}\hat{\boldsymbol{\beta}}^*_{\mathrm{PCA}} = \boldsymbol{x}_{\mathrm{new}}\hat{\boldsymbol{\beta}}_{\mathrm{PCA}}. \tag{7}$$

Algorithm 1: PLS regression

Data: training set $(\mathbf{X}, \mathbf{y})$, number of components k
Result: estimated regression coefficients $\hat{\boldsymbol{\beta}}_{PLS}$

for $i \leftarrow 1$ **to** k **do**
  $\mathbf{w}_i \leftarrow \mathbf{X}^T\mathbf{y} / \sqrt{\mathbf{y}^T\mathbf{X}\mathbf{X}^T\mathbf{y}}$;
  $\mathbf{t}_i \leftarrow \mathbf{X}\mathbf{w}_i$;
  $q_i \leftarrow \mathbf{t}_i^T\mathbf{y} / (\mathbf{t}_i^T\mathbf{t}_i)$;
  $\mathbf{p}_i \leftarrow \mathbf{X}^T\mathbf{t}_i / (\mathbf{t}_i^T\mathbf{t}_i)$;
  $\mathbf{X} \leftarrow \mathbf{X} - \mathbf{t}_i\mathbf{p}_i^T$ (data deflation step);
  $\mathbf{y} \leftarrow \mathbf{y} - \mathbf{t}_i q_i$ (data deflation step);
end
$\mathbf{W} \leftarrow (\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_k)$;
$\mathbf{P} \leftarrow (\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_k)$;
$\mathbf{q} \leftarrow (q_1, q_2, \ldots, q_k)^T$;
$\hat{\boldsymbol{\beta}}_{PLS} = \mathbf{W}(\mathbf{P}^T\mathbf{W})^{-1}\mathbf{q}$

Algorithm 1. PLS regression.

2.4.3 Partial least squares regression

Partial least squares (PLS) regression is very popular in chemometrics (Wold et al., 2001). Similarly to PCA, the input data are transformed, but instead of maximizing the variance of the input data (as in PCA), this transformation maximizes the covariance between input variables and the target. There is no convenient analytical solution for optimization; instead an iterative optimization is employed for parameter estimation. The procedure is presented in Algorithm 1. Here, k is a user-defined parameter such that $1 \le k \le r$; if $k = r$, then PLS regression becomes the ordinary regression.

Nowcasting on new data $\boldsymbol{x}_{\mathrm{new}}$ can be made as

$$\hat{y} = \boldsymbol{x}_{\mathrm{new}}\hat{\boldsymbol{\beta}}_{\mathrm{PLS}}. \tag{8}$$

2.5 Estimating the robustness of linear regression models to missing data

For a linear regression model, it is possible to determine theoretically how many missing inputs can be tolerated before model outputs become obsolete. We can estimate the robustness of a linear regression model to potentially missing input data by using the deterioration index (Žliobaitė and Hollmén, 2013), which is defined as

$$d = -\boldsymbol{\beta}^T(\Sigma - \mathbf{I})\boldsymbol{\beta}, \tag{9}$$

where $\boldsymbol{\beta}$ is a vector of the regression coefficients, assuming that the input variables have been standardized to zero mean and unit standard deviation; Σ is the covariance matrix of the input data; and $\mathbf{I}$ is the identity matrix. High values of the index d indicate low tolerance of the model to missing data. The prediction errors will increase quickly with the number of missing inputs. The smaller d, the more robust to missing data the model is. d may be negative; that is the best option.

Low d guarantees robustness to missing data, but the models with low d do not necessarily give good predictions when

all the values are available. Hence, a tradeoff between accuracy and robustness needs to be found, and the following method can help to find it.

Suppose we get two models A and B, and we would like to select one for deployment. We can measure their prediction errors on a training data set using cross-validation: $\mathrm{RMSE}^{(\mathrm{A})}$ and $\mathrm{RMSE}^{(\mathrm{B})}$. We can also compute deterioration indices $d^{(\mathrm{A})}$ and $d^{(\mathrm{B})}$. Without loss of generality, assume that $\mathrm{RMSE}^{(\mathrm{A})} \geq \mathrm{RMSE}^{(\mathrm{B})}$; i.e., model B shows a better prediction accuracy when no data are missing. If $d^{(\mathrm{A})} \geq d^{(\mathrm{B})}$, then model B is also more robust. In such a case, model B is better (or at least as good as A) with regard to both characteristics, and hence B is preferred over A.

If, however, $d^{(\mathrm{A})} < d^{(\mathrm{B})}$, then we can find out how many input readings can go missing before A becomes better than B. The number m^* can be computed as (Žliobaitė and Hollmén, 2013)

$$m^* = (r-1)\frac{[\mathrm{RMSE}^{(\mathrm{A})}]^2 - [\mathrm{RMSE}^{(\mathrm{B})}]^2}{d^{(\mathrm{B})} - d^{(\mathrm{A})}}, \quad (10)$$

where r is the number of input sensors.

2.6 Experimental protocol

2.6.1 Data preparation and preprocessing

Solar-radiation readings (target variable) are available 99 % of the time. We eliminate from the experiment the samples where no target value is available, since we can use such samples neither for model training nor can we measure the model accuracy on them.

The following preprocessing of the target values is performed. If the measured solar radiation is negative, it is set to 0. If the measured solar radiation exceeds the theoretical (maximum) radiation, the measurement is corrected to be equal to the theoretical radiation. In practice, such observations can arise if, during a cloudy day, the sky is clear where the sun is shining but there is cloud cover elsewhere. The cloud reflects back more back-reflected radiation that the blue sky. For simplicity, we do not consider this effect in our modeling at this stage.

Exploratory analysis of missing data is performed on all 7 years of data. For the analysis of the model accuracies, we use the first 3 years of data as a training set and the remaining 4 years as the testing set. We assume the scenario where an analyst is currently at the end of year three, and all the previous 3 years of data are available for model calibration. After modeling and calibration are done, an online operation scenario is assumed, where the testing data (4 years) arrive in the sequential order.

From the training set we eliminate all the observations that contain any missing values (34 % of train data). The testing set contains all samples, regardless of whether any values in the input data are missing. In addition, we eliminate from the training and testing sets all the observations where the value of theoretical radiation is 0 (the periods of dark) since the value of the target variable is then also 0, which can be nowcasted with 100 % accuracy, while when performing experimental comparison of models we are interested in accuracies of nontrivial nowcasting tasks.

The training data are standardized to have zero mean and unit standard deviation. The testing data are preprocessed by subtracting the mean and dividing by the standard deviation calculated on the training set. After standardization we replace all the missing values in the testing set by zeros and test the performance of the regression models.

Table 2. Summary of regression models: OLS – ordinary least squares; RR – Ridge regression.

		Optimization	
		OLS	RR
Inputs	all r	ALL	rALL
	Selected k	SEL	rSEL
	PCA k	PCA	rPCA
	PLS k	PLS	

2.6.2 Regression models used in the experiments

We experimentally analyze seven regression models, summarized in Table 2.

ALL uses all r sensors as inputs. SEL selects k sensors that have the largest absolute correlation with the target variable (correlation is measured on the training data) and builds a regression model on those k sensors. PCA rotates the input data using principal component analysis, k features corresponding to the largest eigenvalues are retained, and then PCA builds a regression model on those k new features. PLS rotates input data to maximize the covariance between the inputs and the target. We keep k new features.

ALL, SEL, and PCA use the ordinary least squares optimization procedure (OLS) for parameter estimation. In addition, we test the same approaches but using the regularized Ridge regression (RR); these models are denoted as ALLr, SELr, and PCAr. PLS uses its own iterative optimization procedure, which is not regularized.

In addition, we compare the performance to a naive baseline NAI, which produces a constant output, considering that the radiation will be the same as the mean radiation in the training data.

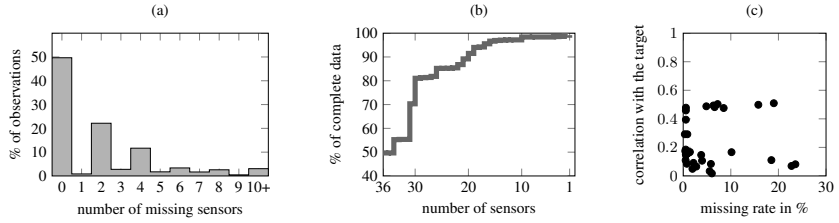

Figure 2. Analysis of missing-data patterns: **(a)** distribution of number of missing sensors in observations (10+ means that 10 to 36 sensors are missing); **(b)** effects of removing the most of the missing sensors; **(c)** the relation of individual sensors with the target variable (each dot represents one sensor).

2.6.3 Software and hardware

The experiments are performed in MATLAB 2012b, using in-house produced code (no extra packages are required) on a commodity laptop computer (Processor 2.5 GHz Intel Core i5; Memory 8 GB 1600 MHz DDR3). The data set used in this study and the code for the experiments are made available[4] for research purposes.

3 Results and discussion

3.1 Analysis of missing-data characteristics

Firstly, we analyze in what way missing values occur in the case study data set. Figure 2a presents the distribution of missing sensors. We see that about half of the time nothing is missing and half of the time observation vectors are incomplete. Over 35 % of the time, 2–4 sensors are missing. The mean number of missing sensors over all the data set is 2.4. We observe from the data that up to 36 sensors (all the input sensors) may be missing at a time. From this analysis we conclude that the amount of missing data is at a massive scale and scope, and missing values needs to be taken into consideration when building nowcasting models on this data. The amount and frequency of missing data also indicates that a case deletion approach would not be suitable because there would be predictions missing continuously.

One may consider that removing from the data set one or two sensors with the largest amount of missing values could solve the problem. This could help if mostly the same sensors were missing all the time. In the following experiment we analyze in what way individual sensors are missing. First, we remove a sensor with the most missing values from the data set; this way the observation vectors at each 30 min time stamp become shorter, and they now include 35 sensors instead of 36. Given the updated observation vectors, we recalculate how many of those vectors contain at least one missing value. Then we remove the sensor lacking the next highest number of measurements and repeat the computation. Figure 2b presents the results. We see that removing a couple of largely missing sensors does not make the remaining observations complete. We would need to remove about half of the sensors in order to reach the stage where at least 95 % of the data are complete. The problem with such an approach is that the removed sensors may carry important information about the target, which then would be lost. To investigate this effect, Fig. 2c presents the relation between the missing-data rate in each sensor and the information about the target contained in it, measured as the absolute linear correlation with the target variable. We have removed the periods where the value of the target is equal to 0 (the dark periods when there is no solar radiation) from this analysis. We see that some sensors in the far right corner and upper center have a high missing-value rate but also high correlation with the target variable. This means that excluding sensors with high missing-value rates would lead to losses of valuable information about the target that would be useful for nowcasting.

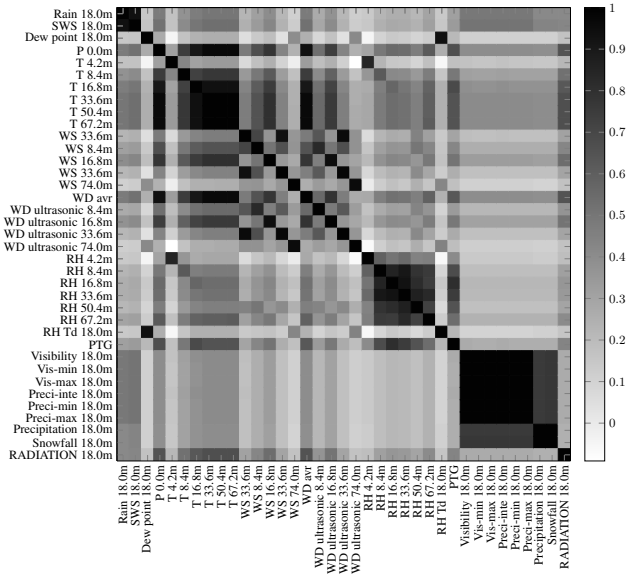

Figure 3. Correlation of missing-value patterns. High correlation (indicated by darker values) means that the values are often missing together.

[4] http://users.ics.aalto.fi/indre/smear.zip

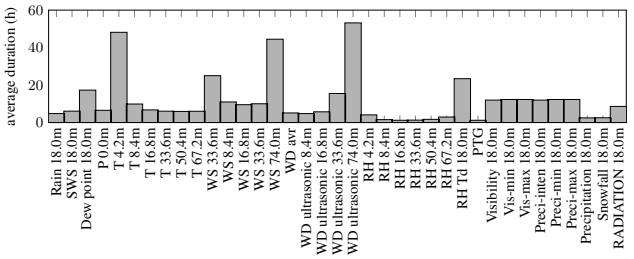

Figure 4. Average duration of missing readings.

One more issue with the data is that sensors do not produce missing values independently of each other. For example, if one temperature value is missing, then it is likely that the other temperature values are missing as well. It may be the case that sensors are missing together due to some common external reasons, for instance, electric power outages. This observation is illustrated by Fig. 3, which plots pairwise correlations between missing values for different sensors. Sensors that are often missing together are encoded in black (dark). We see that, in particular, temperatures (T), relative humidity (RH), visibility, and precipitation readings are often missing together. This means that we cannot rely on the redundancy of the sensors such that, if, say, a temperature reading is missing at 33 m, we can use the reading at 50 m. Both readings would often be missing together.

Finally, in many cases the average duration of missing values lasts for several hours. Figure 4 presents the average duration of missing values in the case study data set for each sensor. Since values may be missing for extended periods of time, we also cannot, from this perspective, simply discard data with missing values, since in such cases we would often not have model outputs for extended periods of time.

In summary, the amount of missing data is very large, and at this level data with missing sensors cannot be discarded without losing valuable information. Missing values are strongly correlated with each other; this makes it difficult and, in many cases, impossible to make use of sensor redundancy or impute missing data based on non-missing data. Removing sensors with the most missing data is also not feasible since missing values are not concentrated in several sensors but are distributed across all the sensors and the sensors with a lot of missing values at the same time carry relatively strong information about the target at times when the values are not missing. Hence, the most appropriate solution to the problem of missing values in this setting appears to be building models that are robust to missing data. This approach is free from any assumptions about the missing data and allows nowcasting even when all or nearly all the sensors are missing.

Table 3. Tenfold cross-validation errors (RMSE) measured based on the training data set and deterioration index (d).

	ALL	rALL	SEL	rSEL	PCA	rPCA	PLS
RMSE	19.0	21.6	20.5	21.9	21.7	21.8	20.8
d	1 122 710	537	362 708	451	−109	−117	6121

3.2 Prediction accuracy

Next we experimentally analyze accuracies of several linear regression models and their robustness to missing values. The first experiment demonstrates how we can select the best model for deployment. The second experiment presents evidence about the performance on unseen data.

Table 3 presents the errors of the regression models ALL, rALL, SEL, rSEL, PCA, rPCA, and PLS, measured on the training set using a fivefold cross-validation and deterioration index estimated based on the training set. For PCA, rPCA, and PLS the number of components was fixed to $k = 18$, which is a half of the original number of input sensors and explains 99 % of the variance. The cumulative percent variance method was used for selecting k, which is recommended as one of the most reliable methods in the literature (Valle et al., 1999). Figure A1 in the Appendix provides complementary information about the variance explained by PCA components. Later in this section we will present a sensitivity analysis to different values of k.

This analysis is performed from the perspective of an analyst, making a decision on which model to deploy. Cross-validation is used to avoid potential overfitting of the model parameters to the training data. Complementary information on the goodness of fit is presented in Appendix B.

In the case of the analyst basing the decision only on the offline analysis of validation errors, he or she would select ALL for deployment since it gives the lowest error, while PCA and rPCA show nearly the highest error. However, the deterioration indexes computed for these models suggest the opposite: rPCA shows the best, while ALL shows the worst deterioration index value. The analyst can now theoretically compare the robustness of two models, for instance ALL and PCA, using the criteria from Eq. (10), which gives

$$m^* = (r-1)\frac{[\mathrm{RMSE}^{(\mathrm{PCA})}]^2 - [\mathrm{RMSE}^{(\mathrm{ALL})}]^2}{d^{(\mathrm{ALL})} - d^{(\mathrm{PCA})}}$$
$$= (36-1)\frac{(21.8)^2 - (19.0)^2}{1\,122\,710 - (-117)} \approx 10^{-4}.$$

The result $m^* = 10^{-4}$ means that, if we expect at least one sensor reading to be missing in 10 000 observations, it is better to deploy rPCA than ALL. Recall that in the data about 2.4 sensors are missing on average in every observation. Hence, in this situation it is clearly worth deploying rPCA instead of a standard linear regression, even though the ordinary regression may be more accurate when no data is missing.

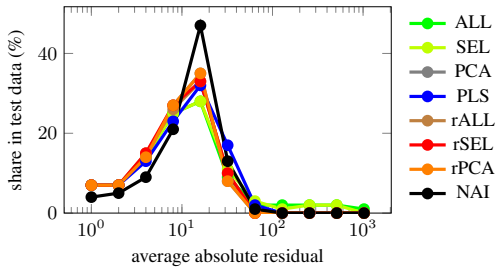

Figure 5. Analysis of residuals.

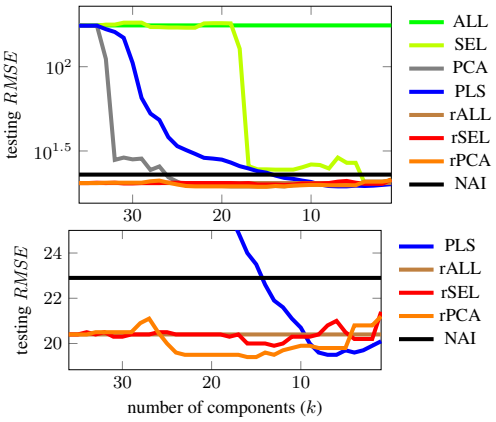

Figure 6. Nowcasting error as a function of components retained (top plot – all models in log scale; bottom plot – best models zoomed in).

Let us consider the regularized version of ordinary regression rALL and rPCA. rALL shows better cross-validation accuracy than rPCA regarding the training data and not as bad a deterioration index as ALL. For rALL and rPCA, $m^* = 0.4$, which means that it is still worth deploying rPCA.

The performance of PCA and rPCA seems very similar. For PCA and rPCA, $m^* = 13.1$, which means that rPCA is expected to be more accurate than PCA if more than 13 sensors are missing; this would be quite pessimistic for our case study data, where the mean number of missing sensors is 2.4. Hence, the analysis suggests chosing PCA for deployment.

The following analysis simulates online operation after deployment. Regression models are trained on the training set, and then sequentially tested on the test set. Table 4 reports the testing results of the regression models ALL, rALL, SEL, rSEL, PCA, rPCA, and PLS ($k = 18$).

The regularized principal component regression rPCA demonstrates the best performance on the test data (RMSE = 19.49), closely followed by PCA without regularization (RMSE = 19.52). The other regularized approaches, rSEL and rALL, perform notably worse (RMSE = 20.43 and 20.28) but they still outperform the naive baseline NAI (RMSE = 22.88). The unregularized approaches PLS, SEL, and ALL perform much worse than the baseline and illustrate well the dangers presented by massively missing values.

It is interesting to note that the analyzed strategy combining linear regression with mean replacement (Žliobaitė and Hollmén, 2013) theoretically approaches NAI performance as more values go missing. If all the input values are missing, then the predictor turns into NAI automatically.

To analyze the performance further, we divide the test data into non-missing (44 %) and missing observations (56 %) and inspect the errors on these subsets separately. We see that the performance of all the models is similar when there is no missing data. The ordinary regression ALL has an advantage in accuracy, since it does not discard any information from the input data. However, the non-regularized models (ALL, SEL, and PLS) fail badly when there is missing data, while the regularized rSEL and rALL lose some accuracy but still remain competitive. Both non-regularized and regularized PCA remain nearly unaffected by missing data.

Figure 5 plots the distribution of absolute residuals for each approach. We can see that most of the errors (residuals) are concentrated around 10, which is a reasonably good result keeping in mind that the range of the target variable is from 0 to 100. It means that most of the predictions do not deviate too much from the true values. We can also see that NAI has less probability mass on the left-hand side, where the most accurate predictions are. As expected, intelligent predictors do better than NAI. Only the unregularized approaches ALL and SEL have any probability mass on the far right, which means that they occasionally produce predictions that may exceed the maximum of the true target. We can conclude from this investigation that predictions by most of the approaches are reasonably stable, and outliers in predictions do not pose any major threats.

Next, we analyze the sensitivity of the predictive performance to different parameter settings. So far we used a fixed number of components ($k = 18$) for PCA, rPCA, and PLS and the same number of selected features for SEL and rSEL. Figure 6 shows the testing errors (RMSE) as a function of k.

An important observation can be made from this plot. The regularized approaches rSEL and rPCA perform reasonably well at all variants of the parameter k, while the non-regularized models SEL, PCA, and PLS perform poorly when a large number of components is retained. In such a case the resulting models are still similar to ALL, which uses all the available information. ALL, rALL, and NAI do not depend on the parameter k but are also included for comparison. We also observe that PLS becomes very effective at low k, but there is a risk of setting k incorrectly (e.g., around 25), in which case PLS gives the worst results. Therefore, we instead recommend using rPCA, which gives stable and accurate results even if k is suboptimal.

Finally, we visually analyze model outputs produced by the baseline approach ALL and a robust approach rPCA ($k = 18$). Figure 7 plots four 3-day snapshots from the year 2012: 1–3 January, 1–3 April, 1–3 July, and 1–3 October. It is important to emphasize that, here, the plot shows raw

Table 4. Nowcasting errors (RMSE) on the testing data set.

	ALL	rALL	SEL	rSEL	PCA	rPCA	PLS	NAI
Full set	175.8	20.4	127.8	20.3	19.5	19.5	25.7	22.9
Non-missing	17.9	19.2	18.8	19.7	19.6	19.6	19.3	22.9
Missing	233.4	21.3	169.2	20.7	19.4	19.4	29.7	22.9

outputs of the classifiers in order to better illustrate the effects of regularization, whereas, when calculating numerical errors, we postprocess all the model outputs to fall into the same interval as the original target ([0, 100], where 0 means no irradiance is observed, and 100 (%) means all the theoretically possible irradiance is observed). That is, if the prediction is less than 0, we correct it to 0, and if the prediction is larger than 100, we correct it to 100. This postprocessing makes the baseline classifiers more competitive (and hence is a more prudent way of quantitative evaluation). We see from the figure that the baseline ALL sometimes fails very badly (particularly in the January and April plots), while the outputs of the regularized approach rPCA remain stable. In July there are only a couple situations when ALL shows very poor performance (we see a green inclination on day 2 and a green peak on day 3). In October both approaches perform similarly. Unlike ALL, rPCA perform in a stable manner and does not exhibit extreme failures.

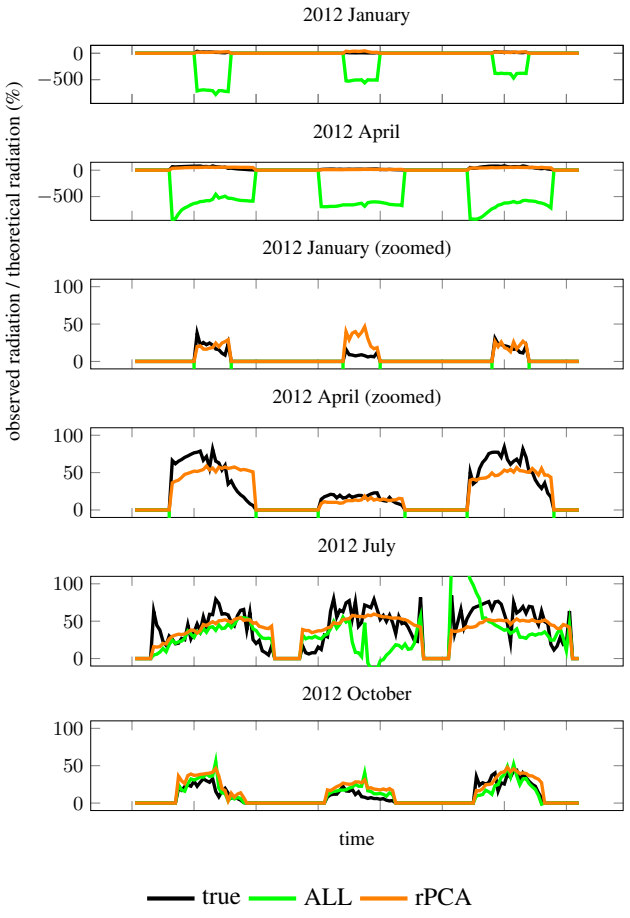

Figure 7. Visualization of nowcasting (each plot shows 3 days).

4 Summary and conclusions

In environmental monitoring, continuous and comprehensive measurement of the environment leads to a data streaming setting. Nowcasting in such settings is a demanding task. We performed a case study in modeling solar radiation based on a SMEAR measurement data set, where model outputs are expected to be available continuously in spite of often missing sensor readings. We also experimentally analyzed missing-data patterns in our data set.

We aimed at nowcasting the amount of global radiation, relative to the theoretical maximum, with the help of measured meteorological variables. Due to the need to provide instantaneous outputs in the data streaming setting, as well as having limited computing power, especially when operating on autonomous power sources, we dismiss all of the sophisticated data imputation methods, which are computationally more demanding. We experimentally analyzed the accuracies and the robustness to missing data of seven linear-regression models and recommend using the regularized PCA regression. The results apply to linear-regression models coupled with the replacement of missing values by a constant (mean).

The strategy that we consider does not require any sophisticated missing-value imputation but just the replacement of the values with predefined constants. Linear regression is also very light computationally; it only requires r multiplications, where r is the number of input variables, and one summation. When the model is trained, it can be recorded and can operate with minimal energy consumption. If, in addition to this, a computationally heavy imputation procedure, such as the expectation maximization algorithm, would require computing power several orders of magnitude greater and would be the dominating computing operation.

A linear regression, supplied with the right input, is a powerful model, particularly considering that, if desired, one could apply nonlinear transformations to the input features, which would then make the resulting predictions nonlinear with respect to the inputs. More importantly, linear models are theoretically well understood and can provide guarantees with respect to performance when there is a lot of missing data. We would argue that in such situations robustness of the model may be more important than flexibility. A flexible model may on average be more accurate, but the outputs may be extremely wrong at times. On the other hand, a robust model may not be the most accurate on average, but its performance would at all times be stable and the errors not too large. We chose linear models since they have theoretical guarantees for robustness. Hence, we recommend using our established guideline in training regression models, which themselves are robust to missing data.

Considering variable uncertainties of sensor measurements over time would make an interesting extension of the current work if we had some way of quantifying how uncertainties vary. The strength of uncertainty could be measured from 0 to 1, where 0 would mean a perfect certainty, 1 would mean a missing value, and everything in between would mean a noisy measurement. In such a case, a missing value could be considered as a special case of uncertainty.

Appendix A: Parameter selection

Figure A1 presents information on the variance explained by PCA components.

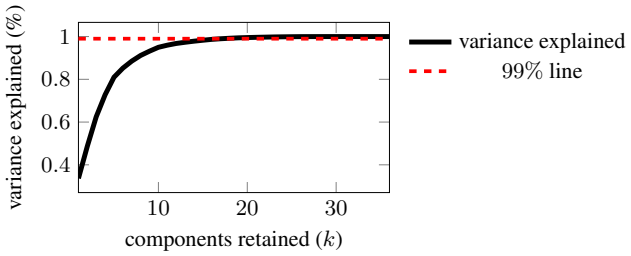

Figure A1. Cumulative variance explained by PCA components on the training data without missing values.

Appendix B: Goodness of fit

Table B1 presents fitness statistics of the regression models to the training data. The coefficient of determination, R^2, indicates the amount of total variability explained by the regression model. The coefficient is computed as

$$R^2 = 1 - \frac{\sum_{l=1}^{n} (\hat{y}^{(l)} - y^{(l)})^2}{\sum_{l=1}^{n} (y^{(l)} - \overline{y})^2},$$

where $y^{(l)}$ is the true target value of the lth sample and $\hat{y}^{(l)}$ is the corresponding model output, $\overline{y}$ is the mean of the true target values, and n is the number of samples in the train set. We see that the best-fit model is ALL. Recalling the experimental analysis in Sect. 3, we can see that good fitness to the training data does not guarantee good generalization performance when a lot of missing values start to appear.

Table B1. Fitness statistics of the models on the training data.

	ALL	rALL	SEL	rSEL	PCA	rPCA	PLS
RMSE	19.2	21.2	20.5	21.6	21.8	21.8	20.9
R^2	0.501	0.393	0.436	0.373	0.361	0.361	0.410

Acknowledgements. This work has been supported by the Academy of Finland grant 118653 (ALGODAN) and grant 258568 (MultiTree).

Edited by: M. Weber

References

Aggarwal, Ch. (Ed.): Data Streams – Models and Algorithms, Springer, 2007.

Allison, P.: Missing Data, Sage Publications, 2001.

Babcock, B., Babu, S., Datar, M., Motwani, R., and Widom, J.: Models and Issues in Data Stream Systems, in: Proc. of the 21st ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, PODS, 1–16, 2002.

Bacher, P., Madsen, H., and Nielsen, H. A.: Online short-term solar power forecasting, Sol. Energy, 83, 1772–1783, 2009.

Bhardwaj, S., Sharma, V., Srivastava, S., Sastry, O., Bandyopadhyay, B., Chandel, S., and Gupta, J.: Estimation of solar radiation using a combination of Hidden Markov model and generalized Fuzzy model, Sol. Energy, 93, 43–54, 2013.

Black, C., Broadstock, D., Colin, A., and Hunt, L. C.: Filling in the gaps in transport studies: a practical guide to developments in data imputation methods, Traffic Eng. Control, 48, 358–363, 2007.

Enders, C. K.: Applied Missing Data Analysis, Guilford Press, 2010.

Hammer, A., Heinemann, D., Lorenz, E., and Lückehe, B.: Short-term forecasting of solar radiation: a statistical approach using satellite data, Sol. Energy, 67, 139–150, 1999.

Hari, P. and Kulmala, M.: Station for Measuring Ecosystem-Atmosphere Relations (SMEAR II), Boreal Environ. Res., 10, 315–322, 2005.

Hastie, T., Tibshirani, R., and Friedman, J.: The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Springer-Verlag, 2001.

Hoerl, A. E. and Kennard, R. W.: Ridge regression: biased estimation for nonorthogonal problems, Technometrics, 12, 55–67, 1970.

Hrust, L., Klaic, Z. B., Krizana, J., Antonic, O., and Hercog, P.: Neural network forecasting of air pollutants hourly concentrations using optimised temporal averages of meteorological variables and pollutant concentrations, Atmos. Environ., 43, 5588–5596, 2009.

Jolliffe, I. T.: Principal Component Analysis, 2nd Edn., Springer, 2002.

Junninen, H., Niska, H., Tuppurainen, K., Ruuskanen, J., and Kolehmainen, M.: Methods for imputation of missing values in air quality data sets, Atmos. Environ., 38, 2895–2907, 2004.

Junninen, H., Lauri, A., Keronen, P., Aalto, P., Hiltunen, V., Hari, P., and Kulmala, M.: Smart-SMEAR: on-line data exploration and visualization tool for SMEAR stations, Boreal Environ. Res., 14, 447–457, 2009.

Kadlec, P., Gabrys, B., and Strandt, S.: Data-driven soft sensors in the process industry, Comput. Chem. Eng., 33, 795–814, 2009.

Kopp, G. and Lean, J. L.: A new, lower value of total solar irradiance: Evidence and climate significance, Geophys. Res. Lett., 38, L01706, doi:10.1029/2010GL045777, 2011.

Lerner, U., Moses, B., Maricia, S., McIlraith, Sh. A., and Koller, D.: Monitoring a Complez Physical System using a Hybrid Dynamic Bayes Net., in: Proc. of the the 18th Conference in Uncertainty in Artificial Intelligence, UAI, 301–310, 2002.

Lu, H., Hsieh, J., and Chang, T.: Prediction of daily maximum ozone concentrations from meteorological conditions using a two-stage neural network, Atmos. Res., 81, 124–139, 2006.

Marquez, R. and Coimbra, C. F.: Forecasting of global and direct solar irradiance using stochastic learning methods, ground experiments and the NWS database, Sol. Energy, 85, 746–756, 2011.

Menut, L. and Bessagnet, B.: Atmospheric composition forecasting in Europe, Ann. Geophys., 28, 61–74, doi:10.5194/angeo-28-61-2010, 2010.

Michalsky, J.: The Astronomical Almanac's algorithm for approximate solar position (1950–2050), Sol. Energy, 40, 227–235, 1988.

Ramoni, M. and Sebastiani, P.: Robust Learning with Missing Data, Machine Learning, 45, 147–170, 2001.

Valle, S., Li, W., and Qin, S. J.: Selection of the number of principal components: the variance of the reconstruction error criterion with a comparison to other methods, Ind. Eng. Chem. Res., 38, 4389–4401, 1999.

Vuilleumier, L., Calpini, B., Cattin, R., Roulet, Y.-A., Stauch, V., Stöckli, R., and Giunta, I.: Solar radiation now-casting: the need for multiple data source integration, in: Proc. of the COST Action ES1002 "WIRE" State of the Art Workshop, available at: http://www.wire1002.ch/fileadmin/user_upload/Major_events/WS_Nice_2011/Spec._presentations/Vuillemier.pdf (last access: 14 July 2014), 2011.

Wold, S., Sjostroma, M., and Eriksson, L.: PLS-regression: a basic tool of chemometrics, Chemometr. Intell. Lab., 58, 109–130, 2001.

Žliobaitė, I. and Hollmén, J.: Fault tolerant regression for sensor data, in: Proc. of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases, ECMLPKDD, 449–464, 2013.

Žliobaitė, I. and Hollmén, J.: Optimizing regression models for data streams with missing values, Mach. Learn., 1–27, doi:10.1007/s10994-014-5450-3, 2014.