Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

Data selection methods for Soft Sensor design based on feature extraction

Riccardo Caponetto*, Graziani Salvatore*
Maria G. Xibilia**

**Dipartimento di Ingegneria Elettrica Elettronica e Informatica, University of Catania, Catania, 95125 ITA (Tel: +390957382327; e-mail: salvatore.graziani@dieei.unict.it)*
*** Dipartimento di Ingegneria, University of Messina, Messina, 98166 ITA (Tel: +390906765328; e-mail: mxibilia@unime.it)*

Abstract: Data selection is a critical issue in data-driven soft sensor design. The paper proposes a new method for data selection based on a feature extraction step, followed by data selection algorithms. The method has been applied to an industrial case study, i.e., the estimation of the quality of processed wastewater produced by a Sour Water Stripping plant working in a refinery. The paper reports the results obtained with different data selection algorithms. The comparison has been performed both by using raw data and the feature extraction phase.

Keywords: Soft Sensors, Model identification, Feature extraction, Data selection. Neural Networks.

1. INTRODUCTION

The monitoring of processes is a significant issue in many industrial applications, where relevant variables need to be measured for either controlling, monitoring, or fault-detection purposes. Acquiring data requires installing measurement equipment, which needs to work online, with a corresponding economic effort, due to hardware and maintenance costs. Eventually, measuring hardware works in harsh environments, with significant risks of failures. A meaningful alternative to conventional instrumentation is using Soft Sensors (SSs). An SS is a software tool that estimates relevant quantities (the SS outputs), as a function of a set of input quantities (the SS inputs). They are used when the output quantities are either hard to be measured, or acquired with too low a sampling time for implementing efficient control or monitoring policies.

Nonlinear data-driven models are widely used in the design of SSs for industrial applications, e.g., Kadleck et al. (2009), Fortuna et. al. (2005), Fortuna et al. (2006), Fortuna et al. (2007), Chen et al. (2011), Yao et al. (2017), Yuan et al. (2017), Yuan et al. (2018), because of the complexity of involved phenomena that hinders the implementation of first-principle models. The SS design relies on the quality of data used in the synthesis phase; the better the data quality, the better the SS performance. It is required that data used in the design phase of the SS represent all the process dynamics. Unfortunately, it is generally hard running experimental campaigns ad-hoc, so that the SS is designed based on historical databases. Usually, input quantities are acquired at a very fast pace, while output variables are acquired only a few times per day. Such a condition occurs, e.g., when input variables are process quantities (such as flows, temperatures, and pressures), and the output variables are quality indicators (such as composition). In such cases, the output quantity is measured by lab analyses on material samples taken from the process. A dramatic reduction of available data results in these cases, named "data scarcity" problem, Fortuna et al. (2009), Napoli et al. (2011), and Fortuna et al. (2007). In such cases, the opportunity to efficiently process the set of available data is even more critical, Fortuna et al. (2007), and suitable approaches, including semi-supervised learning, have been proposed, Shao et al., (2017), Andò et al. (2019), and Graziani et al. (2018).

Neural Networks (NNs) are a common design tool for data-driven SSs, Fortuna et al. (2007). The training of NN-based SSs requires to organize data into learning, validation, and test data sets. Learning and validation data sets are used for the NN training. The test data set is used for estimating the SS performance. The most obvious method for organizing data is randomly picking-up input-output pairs. Nevertheless, techniques exist in literature for more efficient extraction of learning data from the whole set, Singh et al. (2019), and Galvao et al. (2005).

In this paper, the effect of data selection methods on the performance of SSs designed in the presence of the "data scarcity" problem is investigated. More specifically, the DUPLEX and SPXY methods are considered, Singh et al. (2019). Both SSs designed on randomly picked data, and SSs designed by selecting data by using DUPLEX and SPXY have been considered. As a novel contribution, here, the selection methods mentioned above are applied both to the input data and to features extracted from original data. More specifically, features are obtained by using a one-layer denoising autoencoder, Goodfellow at al. (2016), and Yu et al. (2018).

The investigation is performed on data collected from a Sour Water Stripping (SWS) plant, working in a large refinery in Sicily. The Authors have already investigated this process as a case study affected by data scarcity, Graziani et al. (2018). In the following, details about the SS design, performed by using

the proposed selection methods are reported. A comparative analysis of the obtained results is moreover performed.

2. THE DESIGN OF THE SS FOR THE SWS PLANT

The SWS plant processes the wastewater produced by the refinery Gofiner plant. Resulting water is released, as the output of the plant, to the wastewater refinery system. The gas produced by the SWS, as a further output, is fed to the refinery Sulphur Unit. The quality of the SWS output is measured by the concentration of H_2S and NH_3 in the wastewater. A scheme of the process and the variable Tags, is shown in Fig. 1.

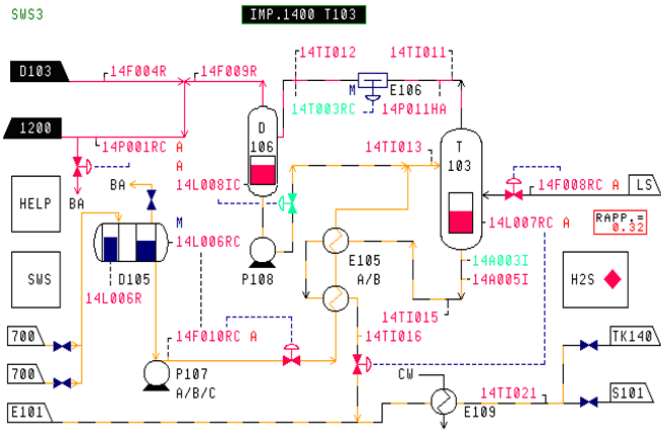

Fig. 1. Scheme of the SWS (T103) and corresponding variable Tags.

The estimation of H_2S by using a SS is investigated in the following. Input variables for the SS design were selected according to suggestions from the SWS plant technologists. The list of selected input and output variables is reported in Table 1. The adopted notation, Tags, and units for the considered variables are also given. Input variables are acquired at a sampling interval of T_s=1 min and averaged. The mean values, in 15 min, are, finally, stored in the database. Data were extracted from the plant database. The plant outputs are sampled once a day and the concentrations of H_2S is obtained offline by laboratory analyses.

The output sampling frequency is too low for process monitoring. An SS is required for estimating the process output quality in real-time. The SS needs to be designed on the basis of a small dataset.

The original dataset refers to a period lasting 1345 d. As a first step, data extracted from the plant database were processed for eliminating outliers. According to plant technologists' suggestions, NaN samples, flat data, and values larger than 30 ppm were removed. Data corresponding to 700 d were available for the SS design.

The proposed SS will assume the form of a static nonlinear MISO model, where the input will contain values of the process variables, and the output will be the H_2S concentration (see Table. 1).

Let consider a MISO (Multi-Input Single-Output) system with p input variables $\boldsymbol{u}=[u_1,\ldots,u_p]$ and one output y.

Table 1. Input and Output Variables, Tags and Units

	Variable description	Tag (Fig.1)	Unit
$\mathbf{u_1}$	Steam to T103	14F008RC	kg/h
$\mathbf{u_2}$	Feed to T103	14F010RC	m^3/h
$\mathbf{u_3}$	BA gas pressure	14P001RC	N/cm^2
$\mathbf{u_4}$	E-106 output flow temperature	14T003RC	°C
$\mathbf{u_5}$	T-103 top steam temperature	14TI011	°C
$\mathbf{u_6}$	T-103 bottom to E-105 temperature	14TI015	°C
$\mathbf{u_7}$	T-103 bottom from E-105 temperature	14TI016	°C
$\mathbf{y_1}$	H_2S content in SWS output	1404A004.AN	ppm

The nonlinear model assumes the form:

$$y(k) = f(\boldsymbol{u}(k)) = f\big(u_1(k)),\ldots,u_p(k)\big), \qquad (1)$$

being $f(\bullet)$ a suitable nonlinear function of the vector $\boldsymbol{u}$. Two-layers Multilayer Perceptrons (MLPs) will be used to this aim.

In the next section, methods used for selecting data for MLPs training. *i.e,* building the learning, validation, and test datasets, will be described.

It is worth mentioning that, though the case study refers to a static model, the proposed strategy can be applied to the more general case of nonlinear FIR or nonlinear ARX models.

3. METHODS FOR DATA SELECTION IN THE DESIGN OF DATA-DRIVEN MODELS

Data selection is one of the main challenges in SSs design and many methods have been proposed, Kennard et al. (2969), Daszykowski et al. (2002), Marengo et al. (1992), Saptoro et al. (2012), and Gao et al. (2018). Nevertheless, many applications are proposed in literature based on the random selection of input data. In Singh et al. (2019), methods for data selection are proposed for the case of large datasets. Here, the same problem is addressed to cope with the problem of data scarcity. More specifically, among possible data selection methods, the DUPLEX and SPXY algorithms are used, as representative of methods that exploit statistics on the input data and input-output data, respectively. Generally, the methods mentioned above are applied to raw input-output data. In this paper, as a novel contribution, the selection methods are applied to a different kind of dataset. More specifically, a preprocessing phase is used for extracting, from raw data, a set of features. These are obtained by using a denoising autoencoder, Goodfellow at al. (2016). The data selection methods are then applied to the feature-output dataset. This allows for exploiting the high-level information contained in the features, which might be more representative than the original data.

The DUPLEX algorithm involves the following steps, Singh et. al., 2019:

- calculate the Euclidean norm, between all possible input data couples and select the couple associated with the maximum value of the norm;
- add the selected couple to the training dataset;
- move the couple associated with the second largest norm value to the test dataset;
- calculate the norms between the two selected couples and all data still contained in the original dataset. The point associated with the maximum distance from the couple in the train set is selected for the training set. The same criterion is applied for the test dataset.
- Continue applying the selection method until all available data are considered.

The SPXY algorithm has been proposed to include statistics of both the input and output variables, Singh et. al., 2019. It is based on the same routine as for the DUPLEX method reported above. The distance is modified as:

$$\tilde{d}_{uy}(z,v) = \frac{d_u(z,v)}{max_{z,v\in D} d_u(z,v)} + \frac{d_y(z,v)}{max_{z,v\in D} d_y(z,v)} \ , \qquad (2)$$

being $d(\bullet,\bullet)$ is the Euclidean norm, D is the dataset, and (z, v) are elements of the dataset.

Both the SPXY and DUPLEX methods have been used in the procedure proposed in this paper. More specifically, these methods are applied to a set of features extracted from the original dataset. The procedure consists of the following steps:

- apply a denoising autoencoder to the raw dataset;
- extract the corresponding features;
- use either the DUPLEX or SPXY method to the feature set, or to the feature set, complemented with the corresponding outputs, respectively, to create the train and test datasets.

The obtained sets are used for the design of SSs by using two-hidden- layer MLPs.

4. NUMERICAL RESULTS AND DISCUSSION

The procedure has been applied to the design of the SS introduced in Section 2. The available samples (700) were split into two sets, each one containing 50% of the original data. It is worth mentioning that this could not be the most effective choice when small data sets are available. Nevertheless, such a choice aimed at evaluating the improvements that can be achieved by using the proposed selection method. For the sake of comparison, SSs, based on MLPs have been trained on the datasets obtained by the following algorithms:

1) random selection from the raw dataset;

2) DUPLEX algorithm applied to to the raw data;

3) SPXY algorithm applied to the raw data;

4) random selection from the feature dataset;

5) DUPLEX algorithm applied to the feature dataset;

6) SPXY algorithm applied to the feature and output dataset.

The same numbering is adopted in the following of the paper when discussing the SSs performance.

In order to extract the features, the same autoencoder, with five hidden neurons (i.e., five features) has been trained. Also, for each MLP topology, the same initial weight matrices have been used for all the algorithms. An exhaustive search has been performed for the MLP structure, with hidden neurons spanning in the range 3 to 10 neurons, for each hidden layer.

The SS performance has been evaluated by using both the Correlation Coefficient (CC) between the measured and estimated output value, and the Root Mean Square Error (RMSE) of the model residual. The coefficients have been evaluated both for the training and test data sets. In the following, the results of such an analysis are given in graphical form from Fig. 2 to Fig. 9. More specifically, the box plots of the CC and RMSE are reported for all mentioned methods. Each box plot reports the results obtained for all considered MLPs.

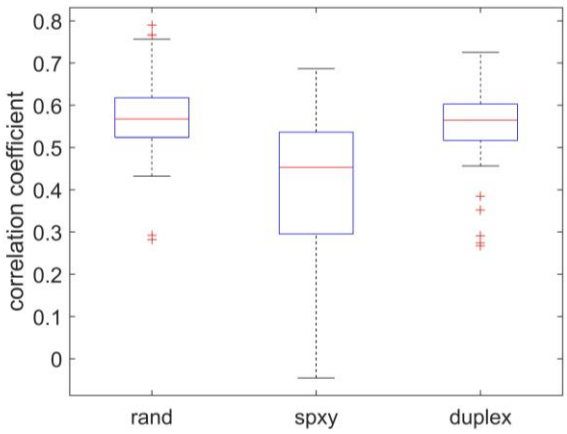

Fig. 2. CC for SS obtained with the methods 1), 2), and 3) on the learning data.

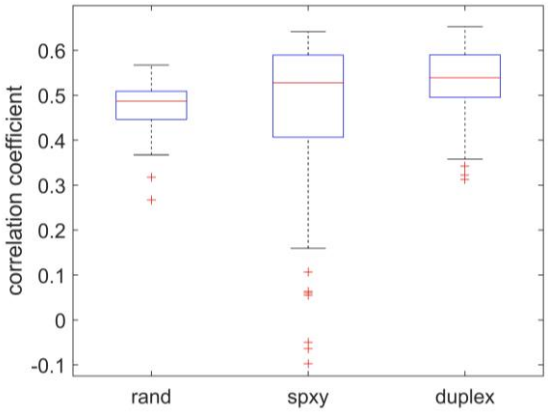

Fig. 3. CC for SS obtained with methods 1), 2), and 3) on the test data.

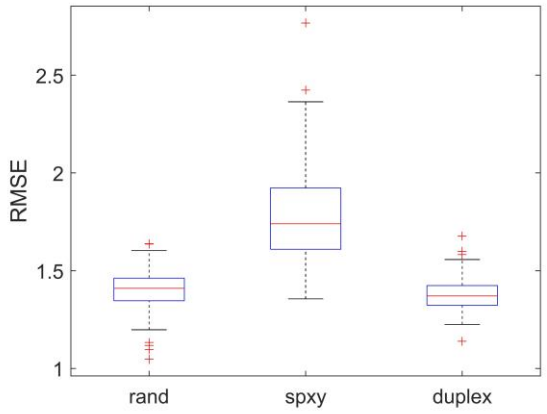

Fig. 4. RMSE for SS obtained with methods 1), 2), and 3) on the learning data.

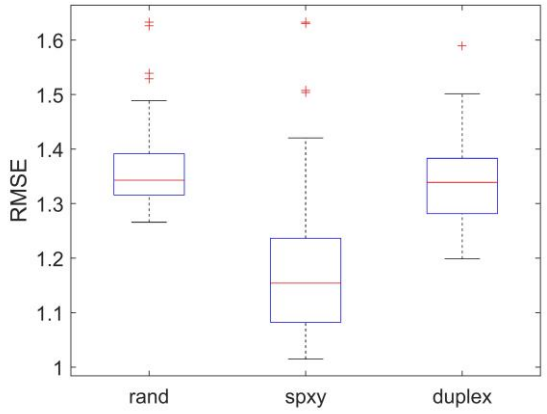

Fig. 5. RMSE for SS obtained with methods 1), 2), and 3) on the test data.

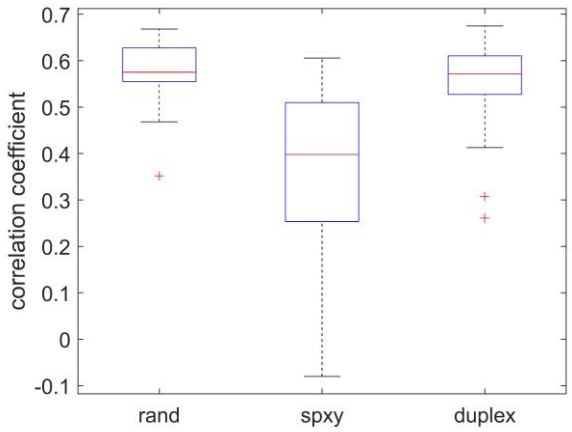

Fig. 6. CC for SS obtained with methods 4), 5), and 6) on the learning data.

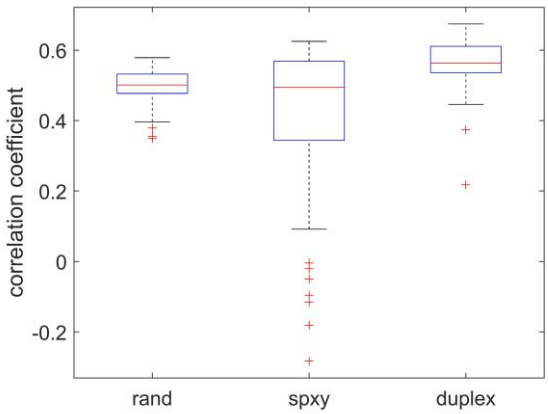

Fig. 7. CC for SS obtained with methods 4), 5), and 6) on the test data.

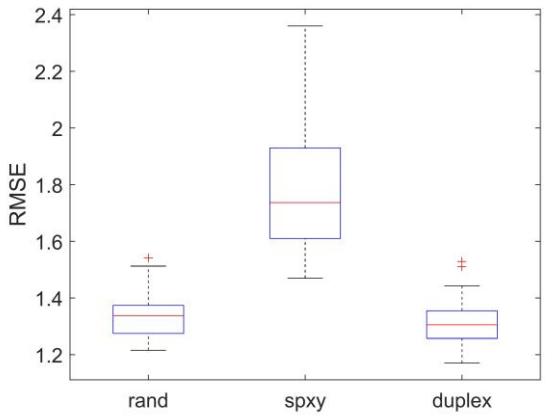

Fig. 8. RMSE for SS obtained with methods 4), 5), and 6) on the learning data.

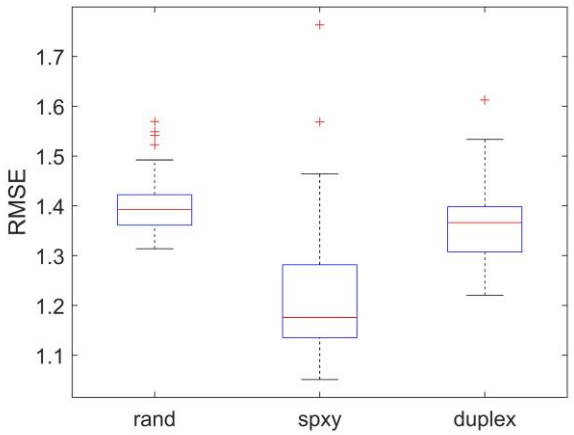

Fig. 9. RMSE for SS obtained with methods 4), 5), and 6) on the test data.

Most relevant parameters, extracted from results reported in Fig. 2 to Fig. 9, i.e., the median values of the CC and RMSE, the maximum CC value, and the minimum RMSE value are reported in Tab. 2 and Tab. 3, for the CC and RMSE, respectively.

From Figs. 2, 3, 6, and 7, relating to the CC, it can be observed that a beneficial effect was obtained on the generalization properties of the SS (test data) by using the DUPLEX algorithm. This effect exists both when the DUPLEX method is applied to the raw data or the features. Moreover, this effect, both in terms of the median and the dispersion of values, is more evident in the second case (see Fig. 7).

The SPXY works worse than random selection on the learning data set, but overperforms the random selection method on the test data.

Table 2. CC values on the learning and test datasets, for all methods

	Learning		Test	
	median	**Max**	**median**	**Max**
1	0.568	0.790	0.487	0.567
2	0.453	0.687	0.528	0.641
3	0.565	0.725	0.540	0.653
4	0.575	0.668	0.501	0.579
5	0.398	0.606	0.469	0.625
6	0.572	0.675	0.563	0.674

From Figs. 4, 5, 8, and 9, relating to the RMSE, it can be observed that, looking at the learning data, the data preprocessing with the autoencoder, generally improved the performance of the SS. The random selection and the DUPLEX algorithm are comparable, though DUPLEX is slightly better working. Looking at the test data, the SPXY has smaller values of the RMSE, but much more dispersed values have been obtained. The DUPLEX works slightly better than random-selection method, both with and without autoencoder data preprocessing.

It is worth noticing that using the DUPLEX algorithm, one network was obtained, guaranteeing both the largest CC value and the smallest RMSE. This consistency was not obtained by using the SPXY method.

As a final remark, though the improvement of SSs performance required an increase of the computation load, the most part of the load characterizes the SS design phase. The computational load of the on-line working is not significantly increased.

Table 3. RMSE values on the learning and test datasets, for all methods

	learning		test	
	median	**min**	**Median**	**min**
1	1.410	1.047	1.343	1.266
2	1.740	1.356	1.154	1.015
3	1.371	1.139	1.339	1.199
4	1.377	1.216	1.392	1.314
5	1.737	1.470	1.176	1.051
6	1.306	1.170	1.366	1.220

Based on results discussed so far, the DUPLEX method, applied to the features, looks the most adequate for designing a SS for the SWS. More specifically, the best results have been obtained by an MLP with 10 and 6 neurons in the hidden layers.

The time plot of the H_2S concentration values and the corresponding estimation are reported in Fig. 10. More specifically, the graph reports the original data and the corresponding estimation, in the original time sequence.

For the sake of completeness, the figure also shows the estimation obtained by using the best performing MLP, among those trained with the classical random selection of raw data.

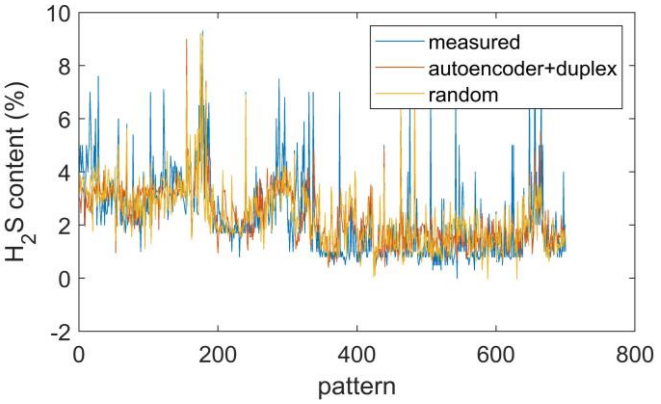

Fig. 10. Time plot of SS estimation of H_2S, obtained with the methods 1) and 5). The plots refer to the whole available dataset.

The proposed selection method (i.e., the autoeconder followed by the DUPLEX algorithm) allowed increasing the CC value from 0,56 to 0,67 and decreasing the RMSE value from 1.27 to 1.22. Such values refer to the test data.

5. CONCLUSIONS

In the paper, methods for selecting data to be used in the design of data-driven SSs have been proposed and compared. More specifically, a new method based on the exploitation of features for data selection is proposed. Features are extracted by using a denoising autoecoder.

The proposed method has been applied to a real industrial case of study, i.e, the estimation of the quality of wastewater produced by a Sour Water Stripping process.

The performance of the proposed method has been compared with the classical random selection method applied to the raw data. Reported results show a beneficial effect on the generalization capabilities of the SS. This reflects in an improvement of both CC and RMSE on test data processing.

Thought the procedure has been applied to a static model, it can easily extended to dynamic nonlinear models, e.g., NFIR or NARX models, by arranging data in suitable time windows.

The proposed method is a general one, and further developments are possible either by using different feature extraction methods and selection algorithms.

REFERENCES

Andò, B., Graziani, S., and Xibilia, M.G. (2019). Low-order Nonlinear Finite Impulse Response Soft Sensors for Ionic Electroactive Actuators Based on Deep Learning. *IEEE Trans. Instr. Meas.*, 68 (5), 1637-1646.

Chen, N., Dai, J., Yuan, X., Gui, W., Ren, W., and Koivo, H.K. (2018). Temperature Prediction Model for Roller Kiln by ALD-Based Double Locally Weighted Kernel Principal Component Regression. *IEEE Trans. Instr. Meas.*, 67 (8), 2001-2010.

Daszykowski, M., Walczk, B., and Massart, D.L. (2002). Representative subset selection. *Anal Chem. Act.,* 468, 91-103.

Fortuna, L., Graziani, S., and Xibilia, M.G. (2005). Virtual instruments in refineries. *IEEE Instr. Meas. Mag.*, 8 (4), 26-34.

Fortuna, L., Graziani, S., Rizzo, A., and Xibilia, M.G. (2006). *Soft sensors for monitoring and control of industrial processes. Advances in Industrial Control*, Springer Verlag, London.

Fortuna, L., Giannone, P., Graziani, S., and Xibilia, M.G,. (2007). Virtual instruments based on stacked neural networks to improve product quality monitoring in a refinery. *IEEE Trans. Instr. Meas.*, 56, 95-101.

Fortuna, L., Graziani, S., and Xibilia, M.G. (2009). Comparison of soft-sensor design methods for industrial plants using small data sets. *IEEE Trans. Instr. Meas.*, 58 (8), 2444-2451.

Galvão, R.K.H., Araujo, M.C.U., José, G.E., Pontes, M.J.C., Silva, E.C., and Saldanha, T.C.B. (2005). A method for calibration and validation subset partitioning. *Talanta*, 67 (4), 736-740.

Gao, T., Hu, L., Jia, Z., Xia, T., Fang, C., Li, H., Hu, L., Lu, Y., and Li, H. (2018). SPXYE: an improved method for partitioning training and validation sets. *Clust. comp.,* 22 (2), 3069–3078.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press, Boston.

Graziani, S., and Xibilia, M.G., (2018). A deep learning based soft sensor for a sour water stripping plant, *Proc. of IEEE I²MTC 2017.* Torino, Italy, 22-25 May 2018, 7969924.

Kadlec, P., Gabrys, G., and Strandt, S. (2009). Data-driven Soft sensors in the process industry. *Comp. Chem. Eng*., 33, 795-814.

Kennard, R.W., Stone, L.A. (1969). Computer aided design of experiments. *Technom.*, 11, 137-148.

Marengo, E., and Todeschini, R. (1992). A new algorithm for optimal, distance based experimental design. *Chem. int. Lab. Syst.,* 16, 37-44.

Napoli, G., and Xibilia, M.G. (2011). Soft Sensor design for a Topping process in the case of small datasets. *Comp. Chem. Eng.*, 35 (11), 2447-2456.

Saptoro, A., Tadè, M.O., and Vuthaluru, H. (2012). A modified Kennard-Stone algorithm for optimal division of data for developing artificial neural network models. *Chem. Prod.Proc. Mod.*, 7, 1-14.

Shao, W., and Tian, X., (2017). Semi-supervised selective ensemble learning based on distance to model for nonlinear soft sensor development. *Neurocomp.*, 222, 15–21.

Singh, H., Pani, A.K., and Mohanta, H.K. (2019). Quality monitoring in petroleum refinery with regression neural network: Improving prediction accuracy with appropriate design of training set. *Measurement: J. Int. Meas.Conf.*, 134, 698-709.

Yao, L., and Ge. Z. (2017). Online Updating Soft Sensor Modeling and Industrial Application Based on Selectively Integrated Moving Window Approach. *IEEE Trans. Instr. Meas.*, 66 (8), 1985-1993.

Yu, J., Hong, C., Rui, Y., and Tao, D. (2018). Multi-task autoencoder model for recovering human poses. *IEEE Trans. Ind. Electr..,* 65 (6), 5060–5068.

Yuan, X., Ge, Z., Song, Z., Wang, Y., Yang, C., and Zhang, H. (2017). Soft Sensor Modeling of Nonlinear Industrial Processes Based on Weighted Probabilistic Projection Regression, *IEEE Trans. Instr. Meas.*, 66 (4), 837-845.

Yuan, X., Wang, Y., Yang, C., Ge, Z., Song, Z., and Gui, W. (2018). Weighted linear dynamic system for feature representation and soft sensor application in nonlinear dynamic industrial processes. *IEEE Trans. Ind. Electr.*, 65 (2), 1508-1517.