Improving Autoencoder Training with novel Goal Functions based on Multivariable Control Concepts

Rafael H. Martello, Lucas Ranzan, Marcelo Farenzena, and Jorge O. Trierweiler

Group of Intensification, Modeling, Simulation, Control, and Optimization of Process, GIMSCOP, Federal University of Rio Grande do Sul, Chemical Engineering Department, 90040-040, Porto Alegre, RS, Brazil (Tel: +55 49-98830-5836; e-mail: {rafael.martello, Jorge.Trierweiler}@ ufrgs.br).

Abstract: Autoencoders are becoming more representative in all fields of knowledge, due to their ability to classify, compress, and identify data patterns. This study objective was to propose entirely new objective functions using multivariable process control concepts as the gain matrix and Relative Gain Array to improve the quality of prediction and classification of an autoencoder. The advantages of the proposed approach are illustrated through a pulp-and-paper industry. The new function results show an improvement in the detection, leading to savings of up to 22 to 38 thousand dollars per month compared to a model using only MSE.

Keywords: Classifier, Neural-network models, Paper industry, Predictive control, Optimization problems

1. INTRODUCTION

The trend of digitalization in the industry, caused by technological advances linked to Industry 4.0, generates large amounts of data in all devices and machines, containing valuable information about normal and abnormal behavior of the process, creating new opportunities for improvement in terms of costs, safety and quality (Sala et al., 2019; Suschnigg et al., 2020). Predictive models are becoming more representative in all fields of knowledge due to their ability to classify occurrences, predict future results, and quantify an output of interest (Ranzan et al., 2020).

Artificial neural networks are mathematical models inspired by the basic structure of a brain. This model consists of several parallel processing units distributed in layers and sometimes called universal approximators, since it is theorized that models can approximate virtually any function to any degree of accuracy (Cybenko, 1989; Hornik et al., 1989). Autoencoders are a type of neural network, which learns in an unsupervised way. In other words, instead of relying on labeled data, we rely on the relationship between the input variables (Charte et al., 2018;). Due to their symmetrical structure, a characteristic of these models is the potential for detecting patterns and anomalies in data. Autoencoders' main applications are data classification and compression (Almotiri et al., 2017; Charte et al., 2018; Martinez-Murcia et al., 2019).

Most industrial processes are multivariable, but with a proper pairing between output and input, there is no significant interaction between channels (Salgado & Conley, 2004). Recently, neural networks applied on multivariable nonlinear systems have shown many successful applications (Zhang *et al.*, 2007; Cong and Liang, 2009). We can compare neural networks to multivariate systems and thus use control concepts to improve the models. Bristol (1966) presented significant results when developing the Relative Gain Array (RGA), quantifying the interaction between channels on the gain (K) of a Multi-Input Multi-Output (MIMO) process.

Autoencoders can be applied to industrial process data to classify, compress, and identify patterns in data, reproducing their inputs with the knowledge acquired by training. Thus combining the knowledge of multivariable control systems with autoencoder models, the objective of this study was to build new objective functions using concepts of the gain matrix (K) and Relative Gain Array (RGA), applying these functions to a case study of a pulp-and-paper industry for classification of sheet-break, seeking to reduce the interaction between the system channels, making the problem more convex and decreasing the chance of stopping at local minima, improve the quality of prediction and classification of autoencoders.

2. CASE STUDY

For the present paper, a real-world dataset obtained from a pulp-and-paper manufacturing industry was used. The dataset was made available by Ranjan et al. (2018) and consists of a multivariate time series, which contains a rare break event that occurs in the industry. The data consists of sensor readings (x's) at regular time intervals of 2 minutes and event labels (y). The main objective of the data is the development of classifier models to predict these break events.

2.1 Problem Description

The multivariate time series is obtained from continuous data streams recorded over time. This kind of data is standard in industrial processes that have several sensors collecting information. The data contains rare unwanted events in the pulp-and-paper process (paper breaks) that should be

prevented. Sensors are placed in different parts of the machine, measuring both raw materials and process variables.

The industrial process described operates in continuous mode. If a break in the paper happens, the process stops, and the resumption may take more than an hour. These sheet-break events cause significant costs for the industry, and even a 5% reduction will give essential cost savings, so we want to predict these events in advance.

The provided data has:

- 18,274 records were collected over 15 days, containing the following 63 columns:
  - **time:** Timestamp of the row (date and time)
  - **y:** Response variable - binary (124 rows (~0.6%) with y = 1 denoting a sheet-break and the rest are y = 0)
  - **x1-x61:** Process variables. All the variables are continuous, except x28 and x61. x61 is a binary variable, and x28 is a categorical variable (Their descriptions are omitted for data anonymity).

3. METHODOLOGY

3.1 Dataset and Preprocessing

The case study objective is to predict sheet-break before it occurs, so, here in this paper, we will try to predict the events 4 minutes in advance, shifting the labels 2 rows up as made by Ranjan (2019).

Before running our algorithm, the dataset needed to be pretreated. Using the binary response variable (y) that had values 1 or 0 correspondings to the sheet-break occurrence (1) or normal process (0), we intended to identify the sheet-breaks minutes before they occur. Therefore, the normal data points corresponding to times up to 4 minutes before a sheet-break was changed from 0 to 1, and the sheet-break point was discarded. The categorical columns (x28 and x61) and the date and time (time) were also removed from the dataset, maintaining a total of 59 variables.

Data were divided randomly into training, validation, and test sets containing 11541, 2924, and 3655 records, respectively. A standardization method was used, transforming to Gaussian data with 0 to 1 variance, and only the subset of data with 0s are used to train the autoencoder model.

3.2 Neural Network

All implementations in this work were performed in Python v.3.6 with Pytorch and Tensorflow to fit the neural networks. The code was implemented through Google Colaboratory. For reproducibility of the results, random seeds were set to (1, 13, 25) in Pytorch and (1) in Numpy.

3.2.1 Autoencoder

The autoencoder model (Fig. 1) selected for the objective function tests is detailed in Table 1 and was obtained through several hyperparameter tests, basing the tests on the model applied by Ranjan (2019). For the present study a simple autoencoder is used to show the applicability of the developed functions, with a future improvement of the model being considered.

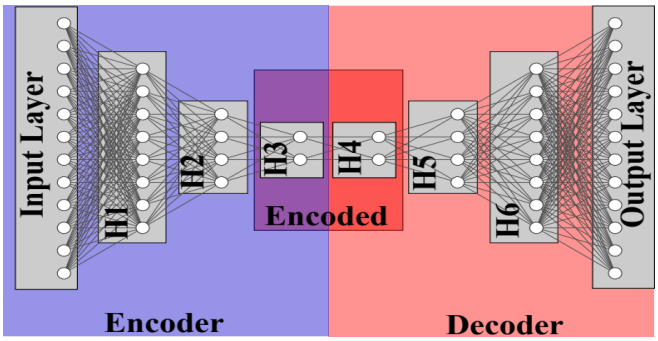

Fig. 1. Autoencoder structure.

Table 1. Model Layers.

Layer	Neurons	Activation
Input	59	ReLU
H1	45	ReLU
H2	32	ReLU
H3	16	ReLU
H4	16	ReLU
H5	32	ReLU
H6	45	ReLU
Output	59	Linear

The model was adjusted using the hyperparameters in Table 2.

Table 2. Model Hyperparameters.

Hyperparameters	Top
Bias	False
Epochs	100
Batch Size	128
Learning Rate	0.001
Optimizer	Adam
Lambda (λ)	0.01

3.2.2 Objective Function

The mean square error (MSE) (1) was used as a base equation for the new objective functions.

$$MSE = \frac{1}{2}\sum_{i=1}^{n}(x_i - \hat{x}_i)^2 \quad (1)$$

3.2.2.1 Gain Matrix (K)

The gain matrix incorporates the input to output variation, composing a matrix between the neural network channels. This matrix comprises the partial derivatives between the channels

following the equivalent definition used in multivariable control systems. It can be calculated by the Jacobian matrix for the variation of all outputs concerning the variation of all inputs, according to equation 2.

$$K = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \cdots & \frac{\partial f_m}{\partial x_n} \end{bmatrix} \quad (2)$$

The matrix K is obtained by the *autograd* Pytorch function, which allows an automatic differentiation between the neural network channel and calculates a derivatives matrix according to equation 2.

Based on the gain matrix, it is proposed the following goal function:

$$f_{gain} = MSE + \lambda(\|K\|_2 - 1) \quad (3)$$

In this goal function, we perform the sum between MSE and the term referring to the model gain, using a lambda(λ) value to define the relative importance of the gain matrix and reducing 1 to correct an ideal system (if the system had identical layers, we would have an identity matrix of norm 1, this being the ideal system or completely linear input to output data). Lambda (λ) is a measure of the relative importance of each function. By increasing the term, we are increasing the penalty for the second term and consequently increasing the loss to bring the system to a more linear state (bringing the weights closer to an identity matrix). The value of 0.01 was chosen based on the work of Hoffman *et al.* (2019).

3.2.2.2 Relative Gain Array (RGA)

Since we are working with square matrices due to the autoencoder having inputs and outputs of similar formats, we can calculate the relative gain array (RGA) that requires inversion and transposition of the gain matrix according to (4),

$$RGA = K \odot (K^{-1})^T \quad (4)$$

where $\odot$ is the elementwise Hadamard product of the two matrices.

In order to reduce the coupling between channels, we created the objective function with the RGA (equation 5), resembling equation 3, using a lambda value(λ) and the norm 2 to define the numerical value of the matrix, decreasing the value of 1 that would correspond to the norm of a completely decoupled model.

$$f_{RGA} = MSE + \lambda(\|RGA\|_2 - 1) \quad (5)$$

3.2.2.2 Jacobian Regularization

Proposed by Hoffman et al. (2019), the function using Jacobian regularization aims to increase the model's robustness, ensuring better results. The model was implemented considering the authors' code, which consists of the *grad* function available in the Pytorch automatic differentiation package (*autograd*). This function accounts for the sum of the variation between channels presenting a vector as an output according to (6) and (7).

$$f_{regJacob} = MSE + \lambda\|J(x)\|_F^2 \quad (6)$$

$$\|J(x)\|_F^2 = \sum_{i,c}\left[\frac{\partial f_c}{\partial x_i}\right]^2 \quad (7)$$

The equation is different from what we use in gain and RGA equations. This model considers the accumulation in each output channel squared using the Frobenius norm.

3.3 Model evaluation metrics

3.3.1 Predictions and class definition

The predictions were verified based on Ranjan (2019), using the model predictions values ($\hat{x}$) for the validation data, compared to the validation input data (x) through mean square error (MSE), evaluating the class with relation to the threshold as class separation parameter, where the error larger than the threshold is equivalent to a sheet-break (1) and error lower than the threshold, normal process (0).

3.3.2 Precision &Recall

Curves of Precision & Recall are helpful metrics in applied machine learning for evaluating binary classification models of unbalanced datasets. The recall represents the ratio of true positives divides by the sum of true positives and the false negatives (8), which can be interpreted as model sensitivity. Precision is a ratio of true positives divided by the sum of true positives and false positives (9), and describes how good a model is at predicting the positive class (Davis and Goadrich, 2006; Koehrsen, 2018).

$$Recall = \frac{TP}{TP+FN} \quad (8)$$

$$Precision = \frac{TP}{TP+FP} \quad (9)$$

The thresholds represent the values that separate the classes (positive and negative). Koehrsen (2018) brings that if we have a model with two classes, the output score will be between 0 and 1, and we can set a threshold in this range for labeling positive and negative classifications. By altering the threshold, we alter the precision versus recall balance.

3.3.3 Receiver Operating Characteristic (ROC) curve

The Receiver Operating Characteristic (ROC) curve is a metric to evaluate performance for classification problems. ROC is a probability curve, and Area Under the Curve (AUC) represents the degree of class separability. The higher the AUC, the better the model predicts the classes.

3.3.4 Reconstruction error and Confusion matrix

Reconstruction error shows the classes True Positive and False Positive above the threshold and the True Negative and False Negative bellow, thus allowing the visualization of error between samples of the dataset and the separation of classes. Confusion matrices have the same end, showing the True

Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN) in the same representation, allowing to verify the values of the predictions numerically. The TP, FP, FN, and TN allow us to calculate accuracy, as shown by Baratloo *et al.* (2015), defined by:

$$Accuracy = TP + \frac{TN}{TP+FN+FP+TN} \quad (10)$$

4. RESULTS

The present study compared the different objective functions under the model developed with the dataset provided by Ranjan et al. (2018). Four objective functions were implemented, MSE, gain matrix function (K), RGA function, and the one developed by Hoffman et al. (2019), which consists of a Jacobian regularization to improve the robustness and stability of the models.

The functions were evaluated for 100 epochs in batches of 128 records. After training, validation data were used to construct Fig. 2, which consists of precision & recall curves for different thresholds and the ROC curves in Fig. 3.

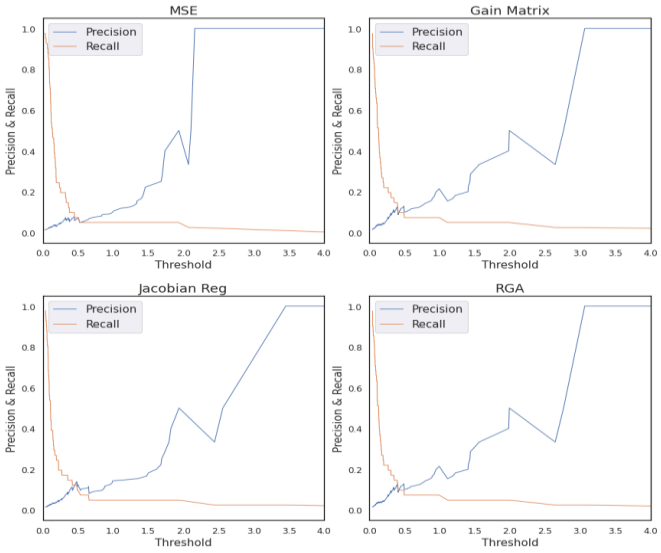

Fig.2. Precision & Recall curve for the four objective functions.

Precision & Recall curves allow us to evaluate the model's quality, with the graphic showing the trade-off between sensitivity (recall) and precision for a model using different thresholds. Relating the thresholds in a binary classification allows us to identify the best threshold point, or the point of best separation between classes, with the highest precision and recall. In this case, the value of 0.4 is used for creating the reconstruction error plot and the confusion matrix. This value corresponds to the intersection point between the recall curve and the precision curve, indicating the highest precision value with the best recall, presenting low values due to unbalanced classes, with the class of interest (TP) presenting a lower number of samples.

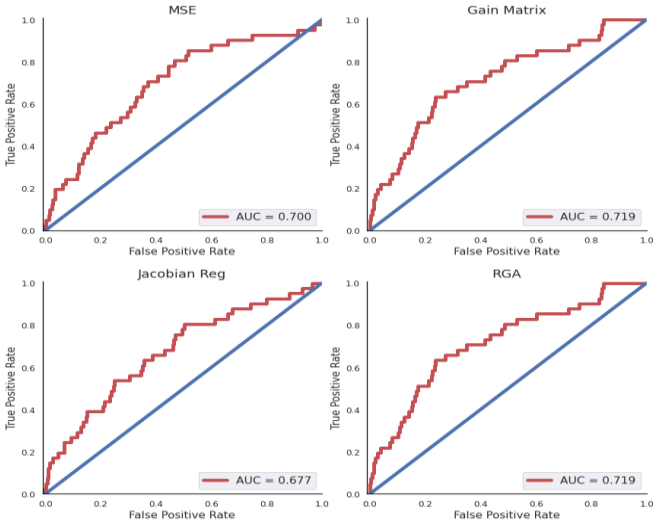

Fig. 3. ROC curves for the different functions.

Evaluating the ROC curves, we can verify an increase in AUC from 0.7 to 0.719 when using the developed functions, indicating a better classification quality of the model.

The reconstruction error representations can be demonstrated in numerical form through confusion matrices (cf. Fig. 4) using pytorch seed 13 to evaluate the function accuracy.

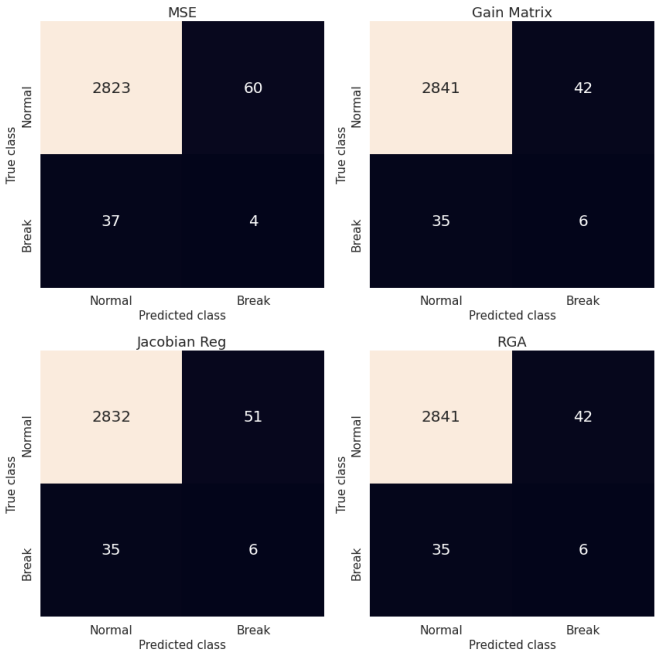

Fig. 4. Confusion Matrix for the different objective functions.

The models trained using the functions developed in the present work have shown a better quality of classification corroborated by the confusion matrices that presented a decrease in the number of false positives and an increase of true positives. Through these data, we can see the potential that the functions presented on an industrial database and its

applicability, and the possible economic gains generated by the work developed. The detection and diagnosis of many of these sheet-breaks root causes would bring substantial value to the industry.

According to Foelkel (2007) and Imtiaz et al. (2007) the sheet-break in pulp-and-paper production is associated with product loss and equipment downtime, which can cause 20 to 96 minutes per day of production stoppage, leading to costs in the range of 11 to 19 thousand dollars per hour stopped, and an annual loss of 6 to 8 million dollars.

The machine learning model allows us to identify rare events 4 minutes before they happen, making it possible to avoid these losses by taking the appropriate actions.

Table 3. Average increment in predictions with different initial weights.

Objective function	True Positive	False positive
Gain (K)	11,1%	-20,97%
RGA	22,22%	-14,51%
Jacobian Regularization	11,1%	1,6%

Using the MSE function, we would have an accuracy of 97.05%, making it possible to avoid 4 out of 41 cases of breakage. Table 3 shows the average increment of the functions detailed in the article for different initial weights in relation to the MSE. Through this table, it is noticed that the functions detailed in the article show an increase of true positives and a reduction of false positives, which generates an improvement in the model classification, with the exception of Jacobian regularization, which increased the number of false positives and the true positive ones, which is reinforced by the reduction of AUC in the ROC curve. The increase in model classification quality leads to savings of 22 to 38 thousand dollars a month using the functions developed.

For the present study, we used random weights according to the Pytorch seeds (1, 13 ,25), to evaluate the Gain/Jacobian matrix behavior, making it possible to visualize, according to Fig. 5, that the matrix tends to a behavior of an identity matrix, thus aiming at reducing the pathways between neurons.

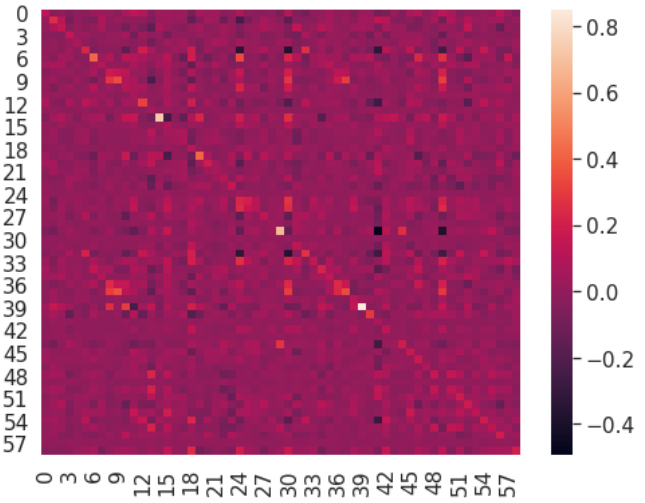

Fig. 5. Post-training gain matrix (K).

The proposed objective functions (Gain and RGA) lead to a more extended training, but an equal prediction time as the MSE, showing an increased recall and precision, indicating an improvement in the predictions of the model.

5. CONCLUSIONS

The present study used multivariable gain concepts and the relative gain array (RGA) to propose two new objective functions for autoencoders' problems. The functions were evaluated by applying the models and functions on a case study of industrial data from a pulp and paper industry to classify leaf breakage problems in the production process. Comparing the results obtained, we improved the detection of 2 true-positive cases and a decrease of 18 false-positive cases. This combined effect is equivalent to a theoretical profit margin of 22 to 38 thousand dollars per month. The functions (Gain and RGA) showed great potential for application, increasing the classification quality for the case study, with better values for precision, recall, and AUC, indicating an improvement in convexity and consequently moving the loss closer to the global minima.

6. ACKNOWLEDGMENTS

We thank the National Center for Scientific and Technological Development (CNPq), the Coordination for the Improvement of Higher Education Personnel (CAPES), and the GIMSCOP Laboratory at the Federal University of Rio Grande do Sul.

7. REFERENCES

Almotiri, J., Elleithy, K. and Elleithy, A. (2017)' Comparison of autoencoder and Principal Component Analysis followed by neural network for e-learning using handwritten recognition', in *2017 IEEE Long Island Systems, Applications and Technology Conference, LISAT 2017*. doi: 10.1109/LISAT.2017.8001963.

Baratloo, A. *et al.* (2015) 'Part 1: Simple Definition and Calculation of Accuracy, Sensitivity and Specificity.',

Emergency (Tehran, Iran). doi: 10.22037/emergency.v3i2.8154.

Bristol, E. H. (1966) 'On a new measure of interaction for multivariable process control', *IEEE Transactions on Automatic Control*. doi: 10.1109/TAC.1966.1098266.

Charte, D. *et al.* (2018) 'A practical tutorial on autoencoders for nonlinear feature fusion: Taxonomy, models, software and guidelines', *Information Fusion*. doi: 10.1016/j.inffus.2017.12.007.

Cong, S. and Liang, Y. (2009) 'PID-like neural network nonlinear adaptive control for uncertain multivariable motion control systems', *IEEE Transactions on Industrial Electronics*. doi: 10.1109/TIE.2009.2018433.

Cybenko, G. (1989) 'Approximation by superpositions of a sigmoidal function', *Mathematics of Control, Signals, and Systems*. doi: 10.1007/BF02551274.

Davis, J. and Goadrich, M. (2006) 'The relationship between precision-recall and ROC curves', in *ACM International Conference Proceeding Series*. doi: 10.1145/1143844.1143874.

Foelkel, C. (2007) 'GESTÃO ECOEFICIENTE DOS RESÍDUOS FLORESTAIS LENHOSOS DA EUCALIPTOCULTURA', *Eucalyptus Online Book & Newsletter*.

Hoffman, J., Roberts, D. A. and Yaida, S. (2019) 'Robust Learning with Jacobian Regularization'. Available at: http://arxiv.org/abs/1908.02729.

Hornik, K., Stinchcombe, M. and White, H. (1989) 'Multilayer feedforward networks are universal approximators', *Neural Networks*. doi: 10.1016/0893-6080(89)90020-8.

Imtiaz, S. A. *et al.* (2007) 'Detection, diagnosis and root cause analysis of sheet-break in a pulp and paper mill with economic impact analysis', *Canadian Journal of Chemical Engineering*. doi: 10.1002/cjce.5450850413.

Koehrsen, W. (2018) *Beyond Accuracy: Precision and Recall*. Available at: https://towardsdatascience.com/beyond-accuracy-precision-and-recall-3da06bea9f6c (Accessed: 16 November 2020).

Martinez-Murcia, F. J. *et al.* (2019) 'Deep Convolutional Autoencoders vs PCA in a Highly-Unbalanced Parkinson's Disease Dataset: A DaTSCAN Study', in *Advances in Intelligent Systems and Computing*. doi: 10.1007/978-3-319-94120-2_5.

Ranjan, C. *et al.* (2018) 'Dataset: Rare Event Classification in Multivariate Time Series'. Available at: http://arxiv.org/abs/1809.10717.

Ranjan, C. (2019) *Extreme Rare Event Classification using Autoencoders in Keras*. Available at: https://towardsdatascience.com/extreme-rare-event-classification-using-autoencoders-in-keras-a565b386f098.

Ranzan, L., Trierweiler, L. F. and Trierweiler, J. O. (2020) 'Prediction of sulfur content in diesel fuel using fluorescence spectroscopy and a hybrid ant colony - Tabu Search algorithm with polynomial bases expansion', *Chemometrics and Intelligent Laboratory Systems*. doi: 10.1016/j.chemolab.2020.104161.

Sala, D. A. *et al.* (2019) 'Multivariate Time Series for Data-Driven Endpoint Prediction in the Basic Oxygen Furnace', in *Proceedings - 17th IEEE International Conference on Machine Learning and Applications, ICMLA 2018*. doi: 10.1109/ICMLA.2018.00231.

Salgado, M. E. and Conley, A. (2004) 'MIMO interaction measure and controller structure selection', *International Journal of Control*. doi: 10.1080/0020717042000197631.

Suschnigg, J. *et al.* (2020) 'Exploration of anomalies in cyclic multivariate industrial time series data for condition monitoring', in *CEUR Workshop Proceedings*.

Zhang, Y. *et al.* (2007) 'Recurrent neural networks-based multivariable system PID predictive control', *Frontiers of Electrical and Electronic Engineering in China*. doi: 10.1007/s11460-007-0037-4.