Article

Analysis of Cleaner Production Performance in Manufacturing Companies Employing Artificial Neural Networks

Rafael Abrantes Penchel 1,* 1, Ivan Aldaya 1, Lucas Marim 1, Mirian Paula dos Santos 1, Lucio Cardozo-Filho 1, Veeriah Jegatheesan 2 and José Augusto de Oliveira 1

1 School of Engineering, São Paulo State University (Unesp), Campus of São João da Boa Vista, São João da Boa Vista 13876-750, Brazil
2 School of Engineering and Water: Effective Technologies and Tools (WETT) Research Centre, RMIT University, Melbourne, VIC 3000, Australia
* Correspondence: rafael.penchel@unesp.br

Abstract: Cleaner production has emerged as a comprehensive paradigm, aiming to reduce, or even avoid, the environmental impact in the production stage, in a broad variety of fields. However, the great number of interacting factors makes the assessment of efficiency and the identification of critical factors pose significant challenges to researchers and companies. Artificial intelligence and, particularly, artificial neural networks have proven their suitability to lead with diverse multi-variable problems, but have not yet been applied to model production systems. In this work, we employ dimensionality reduction in combination with a fully connected feed-forward multi-layer perceptron to model the relation between the input (cleaner production techniques) and output variables (cleaner production performance) and, subsequently, quantify the sensibility of the different output variables on the input variables. In particular, we consider Product Design, Production Processes, and Reuse as the input latent variables, whereas the Environmental Performance of Product, Environmental Performance of Processes, and Economic Performance comprises the output variables of our model. The results, employing data collected from a direct survey of 205 Brazilian companies, reveal that the best configuration for the ANN uses eight neurons in the hidden layer. Regarding sensitivity, the obtained results show that improving practices with poor marks leads to a higher enhancement of output figures. In particular, since reuse presents mainly low marks, it can be identified as an area for improvement, in order to increase overall performance.

Keywords: artificial neural network; cleaner production; environmental performance; economic performance

1. Introduction

Cleaner Production (CP) is the major productive strategy to prevent the environmental impacts in the manufacturing of products [1,2]. CP was introduced in 1990, and has been developed and implemented throughout the globe [3]. Recognized worldwide by the reach of ecoefficiency indicators, the implementation of its practices is strongly stimulated by improvements in economic, environmental, and production performances [4–6]. Although CP has no market certification or accreditation, productive organizations are encouraging the implementation of CP to attain four groups of motivators [5]: legislative and governmental pressures; regulatory pressures from customers; demands from customers; economic opportunities for cost reduction [2].

CP studies can be roughly divided into three major approaches [2]. The first is the development of technologies for application in products and production processes. The second approach focuses on exploring successful case studies from companies that have adopted CP practices. Finally, the third approach deals with the application of surveys to characterize the adoption of CP by companies. Since its foundation, this environmental
production strategy has presented hierarchical levels, organized into groups of practical possibilities that indicate as to which of these groups suggests greater environmental performance [6]. In decreasing scale, these groups comprise product design, productive processes, reuse, internal recycling, and external recycling [3]. In addition, within each group, there is a wide range of practical possibilities to be implemented by companies. With these application levels, CP can help industries improve their product performance, environmental processes, and economic performance. Nevertheless, the success of a CP approach relies on an accurate multivariate decision-making process. In this context, two main model types have been proposed to assist in this decision-making task [1,2,4–19].

On the one hand, maturity models use a segmented, conceptual proposition to evaluate the maturity of a given type of performance, depending on the use of resources employed. The literature that approaches maturity models has converged to some applications, such as the product development process, seen for example in [9,10,13–15,18–21]. This literature also includes applications in management systems’ certification, as can be seen in [4,11,16,22]. Additionally, there is also a search for correlations of input variables (practices) with output variables (performance) by Structural Equation Modeling (SEM) in CP, as cited in studies by [1,2,5]. In brief, the maturity models have been employed, for instance, in order to optimize system performance by identifying the best practices qualitatively [22], and the SEM presents a correlational framework between these constructs and does not allow the establishment of a hierarchical relationship between these variables. Alternatively, Multi-Criteria Decision-Making (MCDM) models have been explored to assess operational practices’ different scales and/or levels and/or maturity. Several types of MCDM have been studied in the literature, highlighting the following methods: PROMETHEE, TOPSIS, ELECTRE, and the Analytic Hierarchy Process (AHP) proposed by [23]. However, none of these works on MCDM adopted CP as an object of study. In addition, these models establish a framework but do not map the input and output variables, making a sensitivity analysis unfeasible.

Since its beginning, one of the main applications of machine learning was to map inputs into outputs without requiring a deep knowledge of the system [24]. Machine learning has been successfully applied in areas as diverse as geosciences [25,26], telecommunications [27–29], medicine [30,31], and the economy [32,33], among others. In this sense, operation management is not an exception. For instance, in [8], an AHP was integrated with an ANN, aiming to improve the operational performance of lean manufacturing, whereas in [6], an AHP was combined with fuzzy logic. On the other hand, logistics was optimized by integrating two MCDM methods with ANN [7] and employing a Gray Decision Model and machine learning [34]. Regarding the environmental performance of production processes, that is, CP, an extensive literature review revealed that machine learning has not been applied for the optimization or identification of the most relevant practices that contribute to each specific performance. In this context, machine learning, and most specifically ANNs, can be employed as a tool to build a map between the input variables, i.e., manufacturing practices, and output variables, i.e., performances.

In the present work, we first develop, optimize, and validate an ANN to model the system, mapping three manufacturing practices (product design, production processes, and reuse) into the environmental performance of the product, environmental performance of processes, and economic performance, considering a database of 205 Brazilian enterprises that practice CP. Afterward, the built map is employed to perform a sensitivity analysis that quantifies the relevance of three manufacturing practices and their impact on each performance metric. The remainder of this article is structured as follows: the collected data and research methods are described in Section 2, whereas results are presented in Section 3. Finally, conclusions are drawn in Section 4.

2. Collected Data and Research Method

2.1. Data Collection and Dimensionality Reduction

The data collection was carried out through a survey applied to \( N = 205 \) (two hundred and five) Brazilian industrial companies. In this collection step, the variables were associ-
ated with representative questions, structured on a semantic differential scale, in which 1 represents “Strongly Disagree” and 7 means “Strongly Agree”. These variables were divided into input variables that are related to the implemented CP practices (adapted from [3], and output variables that quantify the impact of the adopted CP strategies, in terms of the environmental, product, and economic performance of the manufacturing process. The validation of the data collection process is detailed in [2], where SEM was used to establish a correlational framework between the constructs (latent variables). The observable variables, that is, the ones that can be directly obtained from the survey, are denominated manifest variables. In our case, we considered 18 input and 17 output manifest variables. However, the high number of variables may lead to a model with an elevated number of degrees of freedom (DoFs), which would require a prohibitively large data sample to be trained. A widely accepted solution is to combine sets of related manifest variables with latent variables [35]. In this way, the number of input variables was reduced from 18 manifest variables to 3 latent variables, whereas the number of output variables decreased from 17 to 3. In machine learning terminology, the reduction of the amount of input and output variables is denominated dimensionality reduction, and it represents a critical step in many data processing applications. This dimensionality reduction generally uses some correlation or independence analysis to filter out redundant information, leading to more efficient modeling. In our case, the amount of sample data is not sufficiently high to implement sophisticated methods, such as independent component analysis (ICA), so we adopted a simpler approach, in which the latent variable is computed by averaging the associated manifest variables. Table 1 lists the input latent and manifest variables, alongside their respective survey statements. Table 2 lists the input latent and manifest variables, alongside their respective survey statements.

Table 1. Latent input variables, alongside their associated manifest variables and the corresponding survey statement.

<table>
<thead>
<tr>
<th>Latent Variable</th>
<th>Manifest Variable</th>
<th>Survey Statement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Product Design</td>
<td>$p_{d1}$</td>
<td>Our company replaces toxic and/or polluting materials in product design</td>
</tr>
<tr>
<td></td>
<td>$p_{d2}$</td>
<td>Our company makes modifications to the product design in order to improve/adapt the environment</td>
</tr>
<tr>
<td></td>
<td>$p_{d3}$</td>
<td>Our company empowers employees to develop cleaner products</td>
</tr>
<tr>
<td>Production Processes</td>
<td>$p_{p1}$</td>
<td>Our company cleans and organizes the production/shop floor environments</td>
</tr>
<tr>
<td></td>
<td>$p_{p2}$</td>
<td>Our company systematically manages stocks (raw materials/inputs/final products)</td>
</tr>
<tr>
<td></td>
<td>$p_{p3}$</td>
<td>Our company performs equipment maintenance periodically</td>
</tr>
<tr>
<td></td>
<td>$p_{p4}$</td>
<td>Our company improves and standardizes the equipment of the production process</td>
</tr>
<tr>
<td></td>
<td>$p_{p5}$</td>
<td>Our company standardizes work instructions in the production processes</td>
</tr>
<tr>
<td></td>
<td>$p_{p6}$</td>
<td>Our company separates waste and waste from production processes</td>
</tr>
<tr>
<td></td>
<td>$p_{p7}$</td>
<td>Our company has mechanisms for collecting all types of tailings (including spills and burns)</td>
</tr>
<tr>
<td></td>
<td>$p_{p8}$</td>
<td>Our company empowers employees to carry out cleaner production processes</td>
</tr>
<tr>
<td></td>
<td>$p_{p9}$</td>
<td>Our company replaces toxic and/or polluting materials in production processes</td>
</tr>
<tr>
<td></td>
<td>$p_{p10}$</td>
<td>Our company controls the production processes</td>
</tr>
<tr>
<td></td>
<td>$p_{p11}$</td>
<td>Our company makes changes in the production processes</td>
</tr>
<tr>
<td></td>
<td>$p_{p12}$</td>
<td>Our company makes technological changes in production processes</td>
</tr>
<tr>
<td>Reuse (R)</td>
<td>$r_{1}$</td>
<td>Our company reuses waste and residues from a production process as by-products for the same production process</td>
</tr>
<tr>
<td></td>
<td>$r_{2}$</td>
<td>Our company reuses water used in a production process as a resource for the same production process</td>
</tr>
<tr>
<td></td>
<td>$r_{3}$</td>
<td>Our company uses energy from a production process as a resource for the same production process</td>
</tr>
</tbody>
</table>
Table 2. Latent output variables, alongside their associated manifest variables and the corresponding survey statement.

<table>
<thead>
<tr>
<th>Latent Variable</th>
<th>Manifest Variable</th>
<th>Survey Statement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Environmental Performance of Product ($E_{PROD}$)</td>
<td>$e_{prod1}$</td>
<td>The durability of our products</td>
</tr>
<tr>
<td></td>
<td>$e_{prod2}$</td>
<td>The recycling capacity (recyclability) of our products</td>
</tr>
<tr>
<td></td>
<td>$e_{prod3}$</td>
<td>The energy consumption of our products</td>
</tr>
<tr>
<td></td>
<td>$e_{prod4}$</td>
<td>The use of toxic and/or polluting materials in our products</td>
</tr>
<tr>
<td>Environmental Performance of Processes ($E_{PROC}$)</td>
<td>$e_{proc1}$</td>
<td>Air emissions from our production processes</td>
</tr>
<tr>
<td></td>
<td>$e_{proc2}$</td>
<td>The generation of industrial wastewater from our production processes</td>
</tr>
<tr>
<td></td>
<td>$e_{proc3}$</td>
<td>The generation of solid waste from our production processes</td>
</tr>
<tr>
<td></td>
<td>$e_{proc4}$</td>
<td>The consumption of toxic and/or polluting materials and/or substances from our production processes</td>
</tr>
<tr>
<td></td>
<td>$e_{proc5}$</td>
<td>The consumption of electricity by our production processes</td>
</tr>
<tr>
<td></td>
<td>$e_{proc6}$</td>
<td>Water consumption by our production processes</td>
</tr>
<tr>
<td></td>
<td>$e_{proc7}$</td>
<td>The consumption of raw materials by our production processes</td>
</tr>
<tr>
<td></td>
<td>$e_{proc8}$</td>
<td>The frequency of environmental accidents in our production processes</td>
</tr>
<tr>
<td>Economic Performance ($E_{P}$)</td>
<td>$e_{p1}$</td>
<td>The cost of purchasing materials from our company</td>
</tr>
<tr>
<td></td>
<td>$e_{p2}$</td>
<td>Our company’s energy consumption cost</td>
</tr>
<tr>
<td></td>
<td>$e_{p3}$</td>
<td>Our company’s waste treatment rates</td>
</tr>
<tr>
<td></td>
<td>$e_{p4}$</td>
<td>Our company’s waste disposal rates</td>
</tr>
<tr>
<td></td>
<td>$e_{p5}$</td>
<td>Fines for environmental accidents in our company</td>
</tr>
</tbody>
</table>

Once the manifest variables were combined and related to the latent variables, their consistency and reliability were evaluated by the three tests proposed by [35]: Cronbach’s alpha (CA), Composite Reliability (CR), and Average of Variance Extracted (AVE). Regarding CA, for a certain latent variable, the value is given by [35]:

$$CA = \frac{k}{k-1} \left(1 - \frac{\sigma^2}{\sum_{i=1}^{k} \sigma^2_i}\right),$$

(1)

where $k$ is the number of manifest variables associated with the latent variable, $\sigma^2$ is the variance of all the $k$ manifest variables, and $\sigma^2_i$ represents the variance of each manifest variable individually. A value of CA > 0.5 indicates an acceptable consistency. On the other hand, the CR of a latent variable can be calculated as [35]:

$$CR = \frac{\left(\sum_{i=1}^{k} \lambda_i\right)^2}{\left(\sum_{i=1}^{k} \lambda_i\right)^2 + \left(\sum_{i=1}^{k} \delta_i\right)^2},$$

(2)

where $\lambda_i$ and $\delta_i$ are the completely standardized factor loading and the error of the i-th manifest variable, respectively. A value of CR above 0.7 suggests a significant internal consistency of the sample under study. Finally, for a latent variable, the AVE can be computed according to [35]:

$$AVE = \frac{\sum_{i=1}^{k} \lambda_i^2}{k}.$$

(3)

Values of AVE above 0.5 correspond to samples with a significant representativeness. The obtained values for the three latent variables are presented in Table 3, revealing that most of the input and output variables meet the three criteria. The exception is $P_{Prod}$, which falls in two out of the three metrics. This indicates that $P_{Prod}$ may present problems in the model.
Table 3. Reliability and validity of the latent variables.

<table>
<thead>
<tr>
<th>Latent Variable</th>
<th>CA</th>
<th>CR</th>
<th>AVE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Product Design (PD)</td>
<td>0.79</td>
<td>0.88</td>
<td>0.71</td>
</tr>
<tr>
<td>Production Processes (PP)</td>
<td>0.93</td>
<td>0.94</td>
<td>0.81</td>
</tr>
<tr>
<td>Reuse (R)</td>
<td>0.81</td>
<td>0.89</td>
<td>0.73</td>
</tr>
<tr>
<td>Environmental Performance of Product (EPROD)</td>
<td>0.45</td>
<td>0.69</td>
<td>0.34</td>
</tr>
<tr>
<td>Environmental Performance of Processes (EPROC)</td>
<td>0.90</td>
<td>0.92</td>
<td>0.63</td>
</tr>
<tr>
<td>Economic Performance (EP)</td>
<td>0.75</td>
<td>0.85</td>
<td>0.66</td>
</tr>
</tbody>
</table>

2.2. Modeling of Complex Systems Using Artificial Neural Networks

Machine learning (ML) comprises a broad range of algorithms, characterized by a high number of tuning parameters that lead to extremely high flexibility in modeling nonlinear systems [24]. Thus, by properly optimizing the algorithm parameters, ML can model and predict the behavior of complex systems where analytical modeling is difficult or computationally unfeasible. Among the different ML approaches, artificial neural networks (ANNs) have emerged as a powerful tool [36,37]. ANNs emulate the operation of biological neural systems, where different inputs are non-linearly combined. Each neuron on its own performs a simple computation, but when interconnecting a large number of them, the generated ANN acquires a high degree of adaptability, being capable of modeling extremely complex and nonlinear systems.

Recent decades have seen an exponential development and sophistication of ANN architectures and applications. The universe of ANNs can then be classified according to many different characteristics: For example, ANNs can be divided into classifiers and regressors, depending on whether the output domain is continuous or discrete. Another classification criterion is the direction of the information flow. Thus, if the information travels always from neurons closer to the input to neurons closer to the output, the ANN is said to be feed-forward [24]. On the contrary, if the network presents any feedback loops, it is denominated as recurrent. Another possible classification relies on the complexity of the ANN architecture, being possible to have either deep or shallow ANNs. The former uses a huge number of interconnected neuron nodes to process the massive amount of data, including text, audio, and image recognition, or natural language processing. Conversely, the latter employs a relatively low number of neurons to adapt, in order to solve simpler problems. For each ANN architecture, the complexity of the network can be tuned, generally, by adding more connections and neuron nodes. In particular, for a given architecture, a low number of neurons results in a simple model that cannot be adapted to the system, whereas an excessively high number of neurons may lead to a complex model that may fit undesired noisy features. This effect, denominated as overfitting, can be avoided by employing regularization terms in the cost function that penalize high numbers of degrees of freedom, or by some dimensionality reduction. Finally, alongside the architecture, the operation mode of the ANN is another important characteristic of its implementation. ANNs can operate in supervised, unsupervised, or reinforced modes. In supervised mode, the ANN is trained (that is, the weights of each neuron are optimized), in order for the output of the network to be the closest possible to the desired data, which are known a priori. Therefore, a training set of inputs and their associated outputs must be supplied in supervised mode. This is not the case in unsupervised mode, where the weights of the ANN are optimized by employing some kind of metric computed from the output data, without knowledge of the inputs. The reinforced mode can be understood as a hybrid alternative. Typically, initial, supervised learning is performed to achieve a first approximation of the ANN weights. These weights are then progressively refined using the unsupervised mode.

The most suitable architecture and operation mode depends on the problem to be addressed, especially regarding its complexity and available data. For our case, we adopted a swallow feed-forward regressor, operating in supervised mode, to relate the input and output latent variables, as shown in Figure 1. As can be seen, a dedicated ANN is used to
model each latent output variable, allowing for better control of the weights. In addition, this configuration is flexible enough to adapt to relatively low-complexity problems, and does not require a high number of training data. Another advantage of this approach is the low number of parameters to be tuned: the number of neurons in the intermediate hidden layer and the fraction of data dedicated to training. These two parameters must be carefully selected, since they severely impact the modeling and the subsequent sensitivity analysis.

Figure 1. Block diagram showing the relation between input and output latent and manifest variables.

Each of the dedicated ANNs is composed of an input layer, with a number of neurons matching the number of input variables (in our case 3), a hidden layer, with a number of neurons \(N_{HL}\) that has to be tuned, and an output layer, with a single neuron. We used the widely adopted perceptron model [24], which is divided into two stages for artificial neurons. First, the inputs \(x\) are linearly combined by multiplying each input \(x_i\) by the corresponding term \(w_i\) of the weight vector \(w\), and a bias term \(x_0\) is added. In this way, for a neuron with \(N\) inputs, the variable resulting from the linear combination, \(z\), can be mathematically expressed as:

\[
z = w \cdot x + x_0 = \sum_{i=1}^{N} w_i \cdot x_i + x_0. \tag{4}
\]

In the second stage, a nonlinear function \(h(\cdot)\), also called the activation function, is applied to the weighted combination of inputs, giving, as a result, \(y = h(z)\). Different activation functions can be found in the literature, such as the rectified linear unit (ReLU), the arc-tangent, or the sigmoid function [24]. We chose the sigmoid function, because
its gradient has a closed and simple analytical expression, leading to a simple, updating algorithm. The output of the neuron then acquires the form of:

\[ y = h(z) = \frac{1}{1 - \exp(-z)} = \frac{1}{1 - \exp(-w \cdot x - x_0)}. \] (5)

Now, considering the whole network, if the input variables of the \( j \)-th data sample are given by the \( 3 \times 1 \) row vector \( x^{(j)} \), the output vector of the hidden layer \( z_1^{(j)} \) can be expressed in matrix notation as:

\[ z_1^{(j)} = h(W_1 \cdot x^{(j)} + x_{01}), \] (6)

where \( W_1 \) is the weight matrix with size \( N_{hid} \times 3 \), in which the element \((m, n)\) corresponds to the weight of the connection between the \( n \)-th input and the \( m \)-th neuron of the hidden layer. In addition, it is important to note that the bias term \( x_{01} \) is now a column vector formed by the bias of the \( N_{hid} \) neurons. By simple inspection, it is clear that \( z_1^{(j)} \) is a column vector with dimensions \( N_{hid} \times 1 \). Proceeding in the same way for the output layer, the output of the ANN for the \( j \)-th data sample is a scalar given by:

\[ z_{out}^{(j)} = h(w_2 \cdot z_1^{(j)} + x_{02}). \] (7)

Therefore, combining Equations (3) and (4), the output variables can be written in terms of the input variable as:

\[ z_{out}^{(j)} = h(w_2 \cdot h(W_1 \cdot x^{(j)} + x_{01}) + x_{02}) = H_{w_2, w_1, x_{01}, x_{02}}(x^{(j)}). \] (8)

Since our ANN has a single output, the weight matrix becomes a row vector, and the bias term is a scalar one. Optimizing the values for the weights \( W_1 \) and \( w_2 \), as well as the bias terms \( x_{01} \) and \( x_{02} \), in order to reduce the prediction/modeling error, is equivalent to minimizing the following cost function, which depends on both the predicted \( z_{out}^{(j)} \) and the desired value \( y^{(j)} \):

\[ J(W_1, w_2, x_{01}, x_{02}) = \frac{1}{2N} \sum_{j=1}^{N} (z_{out}^{(j)} - y^{(j)})^2 + \lambda \left( \frac{1}{N_{i}N_{h}} \sum_{i=1}^{N_{i}} \sum_{k=1}^{N_{h}} w_{1,i,k}^2 + \frac{1}{N_{o}} \sum_{i=1}^{N_{o}} w_{2,i}^2 \right), \] (9)

where \( N \) is the number of elements in the training data set and \( \lambda \) is the so-called regularization parameter, whereas \( N_i, N_h, \) and \( N_o \) are the number of neurons in the input, hidden, and output layers, respectively. \( w_{1,i,k} \) is the weight of the link between the \( i \)-th input neuron and the \( k \)-th neuron in the hidden layer, and \( w_{2,i,j} \) is the weight of the link connecting the \( k \)-th neuron in the hidden layer to the output neuron. The cost function can be minimized, for example, by error back-propagation, in which the error at the output is propagated toward the input of the ANN, and the weights are updated accordingly. For instance, a detailed mathematical description of the optimization process can be found in [24]. As can be seen, in the calculation of the cost function, all the training data are considered in a single batch. The cost function then depends on the selection of the training data set and, in particular, on the presence of outliers in the training data that may cause the parameters to converge to local minima or to slow down their convergence. This is especially critical when the training set is not as large, as in our case. In order to reduce these effects, for each configuration corresponding to the combination of training set size and the number of neurons, the training was applied to 50 training sets randomly selected from the whole set of data. The rest of the data were used for cross-validation of the model.
2.3. Sensitivity Analysis

Once the ANN had been trained, a sensitivity analysis was performed using a stochastic one-at-a-time (OAT) approach. That is, to assess the impact of an input latent variable on the output latent variables, we slightly increased the value of the input latent variable and computed the difference between the predicted values for the different output latent variables. Since this difference depends on the value of the given input variable and the other input variables, this process was performed for the different input variables. The sensitivity of an output variable $z_{\text{out}}$ in terms of the input variable $x_i$ can then be considered as the partial derivative of $z_{\text{out}}$ with respect to $x_i$ at $x$. Therefore, it can be computed as:

$$\delta z_{\text{out}}, x_i = \frac{\partial z_{\text{out}}}{\partial x_i} \approx H_{w_2, w_1, x_01, x_02} (\delta_i x) - H_{w_2, w_1, x_01, x_02} (x),$$

(10)

where $\delta_i x$ has the same elements of $x$, except for the $i$-th component that has the value of $x_i + \Delta$ instead of $x_i$, with $\Delta$ being a small, constant value.

In Figure 2, we summarize the methodological flow adopted in the present work. The first step is the data collection, conducted in order to obtain a set of original variables. These variables are then reduced via a dimensionality reduction process. Afterward, these processed data are used to train an ANN that models the system. Finally, this model can be used to analyze the sensitivity of the output variables to the input variables. These steps are described in detail in the following subsections.

**Figure 2.** Block diagram showing the relation between input and output latent and manifest variables.

3. Results

3.1. Data Analysis and Dimensionality Reduction

In order to analyze the effect of the dimensionality reduction discussed in Section 2.1, in Figure 3, we show the histograms of the input manifest and latent variables. In particular, Figure 3a represents the histograms of the manifest variables associated with the latent variable of Product Design, whose histogram is shown in Figure 3b; Figure 3c,d correspond.
to the manifest and latent variables associated with Product Process; Figure 3e,f represent the Reuse-related manifest and latent variables.

![Histograms of input manifest variables and the derived latent variables.](image)

**Figure 3.** Histograms of input manifest variables and the derived latent variables. (a) Histogram of the manifest variables associated with Product Design, $P_{PD}$, and (b) histogram of the Product Design. (c,d) Histograms of the manifest and latent variables of Product Process, $P_{Pp}$, and (e,f) Reuse ($R$).

Generally speaking, we can see that each latent variable follows the same tendency as its corresponding manifest variable. For instance, the Product Design and Product Process variables are loaded toward high values, which is similar to the behavior of their respective manifest variables. The Reuse latent variables and their associated manifest variables, on the other hand, show a more flattened distribution. The reader can also perceive that the frequency at the highest scores is reduced in the latent variables, in all three cases. This effect can be explained by noting that, in the histograms of the manifest variables, shown in Figure 3a,c,e, the represented data are integers ranging from one to seven. However, after dimensionality reduction, the corresponding latent variables assume real values, so the histograms in Figure 3b,d,f were built considering seven beams, uniformly spaced, between zero and seven. This explains the apparent reduction of the frequency for high values of the latent variables. In Figure 4, we show similar histograms for the output manifest and latent variables. It should be noted that the manifest variables $p_{P8}$ and $e_{P5}$, both associated with environmental accidents (see Tables 1 and 2), seem to present an anomalous behavior. However, this fact can be attributed to the low number of declared accidents and the relation between this number of accidents and the fines. Once the dimensionality was reduced, both the input and output data were normalized by subtracting the average value of each variable and dividing by the standard deviation. Therefore, each variable has zero mean and unity variance.
Figure 4. Histograms of output manifest variables and the derived latent variables. (a) Histogram of the manifest variables associated with Product Performance, $P_{\text{PROD}}$, and (b) histogram of the latent variable Environment Product Performance ($c,d$) Histograms of the manifest and latent variables of Process Performance, $P_{\text{PROC}}$, and (e,f) Economic Performance ($E_P$).

Before building the ANN, it is important to perform correlation analysis, in order to quantify the correlation between the different input variables and the correlation between the input and output variables. The calculated correlation matrix is graphically shown in Figure 5. As can be observed, considering the input variables, $P_D$ and $P_P$ present some correlation, whereas they are quite uncorrelated with $R$. In regards to the correlation between the input and output variables, the most relevant point is the low correlation between $E_P$ and all three input variables. Consequently, it is expected that $E_P$ will not be related to the input variables when the ANN is implemented.

Figure 5. Correlation analysis.
3.2. System Modeling

Once the manifest variables were grouped into a small number of latent variables, the parameters of the ANN that better fit the relationship between the input and output variables were found. In order to do so, we adopted the Root Mean Square (RMS). Thus, in Figure 6a,b, we present the RMS of the errors of both the training and test sets in terms of the hidden layer size, i.e., the number of neurons in the hidden layer, and the training set size for the environmental performance of the product; in Figure 6c,d we show the RMS of the errors in terms of the environmental performance of the product; in Figure 6e,f we present the same metric, but for the economic performance. Due to the relatively low number of available data, we adopted a k-fold approach to reuse data for test and training. Since this process highly depends on the test–train splitting ratio, we decided to sweep the training set from 100 to 180 elements. In this way, we assessed the performance of configurations with different combinations of neuron numbers and test–train split ratios, which are the two main hyperparameters of our model. In order to reduce the sensitivity to the training and test set partition, for each case, we performed the training and test evaluation of 10 random partitions, and computed the ensemble average of the RMS error values.

![Figure 6](image.png)

**Figure 6.** RMS error, in terms of the hidden layer, and training set sizes for the training and the test sets, for (a,b) the environmental performance of the product, (c,d) the environmental performance of the process, and (e,f) the economic performance.

It is worth mentioning that even if the RMS error of the test set is a priori more relevant than that of the training set, it is important to show the error of both sets, to test whether biasing or overfitting is affecting the modeling. Observing the different plots, it is possible to observe that, in most cases, the RMS error presents high values for low neuron numbers, where the model suffers from biasing. That is, the ANN is too simple to accurately model the relationship between the input and output latent variables. As the number of neurons in the hidden layer increases, the error yields a relatively flat level. Indeed, looking at the
test error, it is impossible to observe a sensitive increase even for a number of neurons in the hidden layer as high as 32, which means no overfitting is present. On the other hand, looking at the dependency of the performance in terms of the training set size, it is possible to observe that the predicted performance tends to be better for larger training sets. This can be explained by the fact that, as the number of elements in the training set increases and the number of elements in the test set decreases, the number of outliers in the latter reduces. Indeed, the presence of outliers in the test set can also explain the non-monotonic behavior in terms of the training set size. The RMS values presented in Figure 6b,d,f reveal that a hidden layer with less than eight neurons incurs biasing, whereas a larger number does not significantly improve the model’s performance. Therefore, a configuration with eight neurons represents a good trade-off between performance and computational complexity.

3.3. Sensitivity Analysis

After finding a suitable combination of ANN parameters, a sensitivity analysis for each output latent variable was performed, according to the method described in Section 2.3. In order to cover all the possible combinations of the input latent variables, we combined two of the input variables, classifying them as good (Note 7), average (Note 4), and bad (Note 1), whereas the value of the third variable was increased from one to seven. In Figure 7, the predicted value for the output variable is represented in color scale, and the variation, i.e., the sensitivity, is proportional to the size of the superimposed circle. The process was carried out for the three output variables. Thus, we present the results for the Environmental Performance of Product in Figure 7a, the Environmental Performance of Process in Figure 7b, and the Economical Performance in Figure 7c. The reader can observe that the sensitivity is generally higher for low input latent variable values. That is, the change in the output latent variable is more significant when the input variables have low marks. In other words, this sensitivity analysis indicates that the enhancement in performance is more notorious if we can improve poorly ranked fields. Furthermore, the sensitivity is almost independent of the value of the latent variable under study, for the three latent variables.

![Figure 7. Sensitivity analysis of Environmental and Economic Performances.](image)

On the other hand, the sensitivity to changes when the other two latent variables have high scores (seven) is relatively lower than when the other two latent variables present low values. In addition, it is worth mentioning that when the other two latent
variables are positively ranked, the sensitivity depends significantly on the mark of the latent variable under study. For example, when we consider the Production process and Reuse with Mark 7, the Environmental Performance of the Product presents a sensitivity in the transition from Grades 1–2 that is much higher than that of 6–7.

Beyond the numerical analysis of the model sensitivity, in terms of the different input latent variables, these results can be interpreted in the context of CP. As expected, when modifying the product design and adapting the production processes, with replacements of raw materials and inputs, good housekeeping and technological innovations, and implementing the reuse of their waste, even at a low magnitude, companies achieve high positive impacts on the environmental performance of the product and process, as well as on the economic performance of manufacturing. Furthermore, from the previous sensitivity analysis, it is possible to conclude that companies' initial implementation of CP significantly impacts performance. As these companies attain a higher degree of maturity of CP, the performance continues to grow, but at a more moderate rate, showing a phenomenon of saturation or decreasing marginal improvement.

In fact, this behavior indicates that in an environment where CP is not practiced, the portfolio of opportunities for environmental improvements in production processes is greater for the first manufacturing projects. On the other hand, when a company has already achieved a relative degree of CP maturity, this portfolio of opportunities is gradually reduced. This relationship is natural and observable in performance measurement systems that follow continuous improvement methods, including CP, which is based on the Plan, Do, Check, and Act (PDCA) approach. The proposed model, therefore, agrees qualitatively with the behavior expected from CP experience. However, at this point, it is important to highlight that the proposed method gives qualitative information on the most critical latent variables, in terms of sensitivity, and quantifies their impact, which can assist the decision-making process and resource management.

4. Conclusions

In this paper, we employed an ANN-based model to quantify the sensitivity of the most critical output latent variables, the Environmental Performance of Product, Environmental Performance of Processes, and Economic Performance, in terms of the input latent variables Product Design, Production Processes, and Reuse.

In order to achieve this sensitivity analysis, we performed a dimensionality reduction on both the input and output variable sets and a sweep of the number of neurons in the hidden layer. When applied to a dataset of 205 Brazilian companies, the model reveals that the output variables are more sensitive to the input variables, when the latter present low scores. That is, it indicates that, for the considered case, improving the input variables that have been poorly ranked leads to a higher enhancement of the output variables.

In addition, it is worth mentioning that the proposed method presents significant potential for reducing the subjectivity of information, which is inherent to data collection based on the opinions of company managers. Furthermore, this model can be adapted to an in loco measurement system of production processes and performance measurement metrics, in light of Industry 4.0, and can be applied to a broad variety of scenarios. Finally, we intend to apply the proposed method to systems in alternative scenarios, in order to assess its reliability and generality, which is a critical step in constructing any model.

Funding: This work was financially supported by Fundação de Amparo a Pesquisa do Estado de São Paulo (FAPESP) grants 2020/11874-5 and 2020/09889-4, Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) grants 313370/2021-5, 409146/2021-8 and 405851/2022-7 and FINEP grant 0527/18.

Informed Consent Statement: Not applicable

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to nondisclosure agreements with the companies.

Conflicts of Interest: The authors declare no conflict of interest.

References


34. Oleśkow-Szląpka, J.; Wojciechowski, H.; Domariski, R.; Pawłowski, G. Logistics 4.0 Maturity Levels Assessed Based on GDM (Grey Decision Model) and Artificial Intelligence in Logistics 4.0-Trends and Future Perspective. Procedia Manuf. 2019, 39, 1734–1742. [CrossRef]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.