IEC
Institute of Electronics and Computer

Journal of Artificial Intelligence and Systems, 2023, 5, 91-114
https://iecscience.org/journals/AIS
ISSN Online: 2642-2859

Seven Machine Learning Methods for Selecting Connecting Rods in the Machining Process

Luciana Claudia Martins Ferreira Diogenes

Frutal - MG, Brazil
Email: lucianafem@yahoo.com.br
*Corresponding Author: Luciana Claudia Martins Ferreira Diogenes, Email: lucianafem@yahoo.com.br

How to cite this paper: Luciana Claudia Martins Ferreira Diogenes (2023). Seven Machine Learning Methods for Selecting Connecting Rods in the Machining Process. Journal of Artificial Intelligence and Systems, 5, 91–114. https://doi.org/10.33969/AIS.2023050107.

Received: September 7, 2023
Accepted: October 10, 2023
Published: October 13, 2023

Abstract

The use of machine learning (ML) has been widely used to control part dimensions during production. Parts manufactured in the automotive sectors also use ML to obtain better accuracy when selecting parts within the customer's specification limit. In this work, it will be simulated how connecting rods can be selected during the machining process inside a metallurgical plant through the application of seven ML algorithms: Decision Tree Classifier, Logistic Regression, k-Nearest Neighbors, Random Forest Classifier, Gaussian NB (Naïve Bayes), SVC (Support Vector Clustering) and Neural Network. The values for the diameter of the small connecting rod eye and the outer diameter are based on data from the literature. Simulations of training and testing data were obtained through Python programming and this data was entered into each of the seven ML techniques.

Keywords

Connecting rods, control, machine learning, production, manufacturing, confusion matrix

1. Introduction

In the functioning of the connecting rod as an important element of an engine that converts piston motion into crankshaft rotational motion can be explain in [1],[2].

Structurally speaking, the connecting rod can be divided into three parts: small end, big end and shank.

The production of connecting rods which can be produced by casting, forging or powder metallurgy method [3]. The forging process is a cheaper process, but the parts do not come out of the forging process with drawing specifications. In this way, the connecting rods are sent to the machining process to reach the specific dimensions. The casting process, on the other hand, generates internal micro-holes in the part and

this compromises the robustness of the product. The powder metallurgy method although more expensive is a process where the dimensions specified in drawings can be easily achieved and it is considered to be a high precision process.

The causes of connecting rod failures were studied in [4]. Failures may be due to structural design, type of material and dynamic loads. The tests used in the study and that can analyze a connecting rod can be visual observation, metallurgical testing, magnetic particle testing, fractography analysis by stereo and scanning electron microscopy, residual stress analysis, dimensional inspection, chemical analysis, Brinell, hardness testing, tensile testing, analysis, microstructure analysis and grain flow analysis.

To optimize the connecting rods, for example, regarding their weight, the Finite Element Method (FEM) can be applied using ANSYS, for example. Weight optimization was performed under a loading force of 30 kN [5].

ANSYS software is used for static analysis of connecting rod also using Finite Element Analysis Method [6]. Simulations are based on the removal of the material from the connecting rods maintaining its strength and integrity.

Failure and stress studies on connecting rods were also performed in [7]. The geometric model was created by CATIA software and later, this model was exported to ANSYS to divide the model into finite elements.

The modal analysis of a connecting rod is presented in [8]. FEM analysis was performed by ANSYS software where they were extracted first six natural frequencies corresponding mode shapes.

The connecting rod model that will be used in this article is shown in Figure 1[6]. The focus will be on simulating small end inner and outer diameter values for both training and testing data.

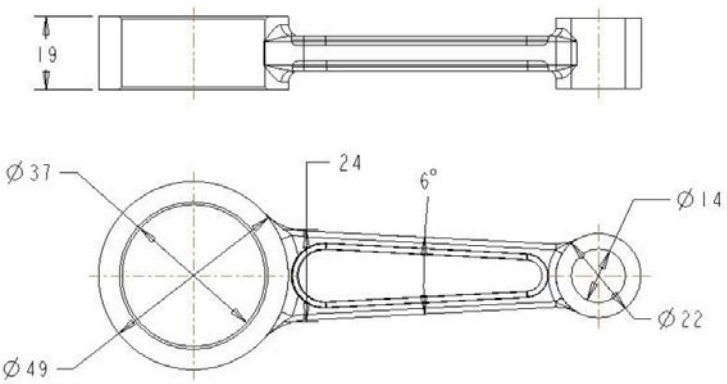

Figure 1. Structure of a connecting rod with its specified dimensions [6].

2. Related Work

Problems of defects and irregularities in parts has always been a problem in manufacturing. As in this work, which portrays defects in the manufacture of connecting rods on a production line using seven machine learning (ML) techniques, other authors commented on the problems of parts manufactured by a process called selective laser sintering (SLS) using only one ML technique [9]. Some tests can be destructive and others non-destructive. These authors applied machine learning (ML) techniques such as convolutional neural networks (CNN) to allow the automatic classification of defects in SLS in non-destructive tests. Powder bed defects can influence the quality of the parts and in some cases, the parts need to be discarded, which in the latter case causes waste of material. The method to evaluate the defect consists in training neural networks with many images. The images of the pieces were captured by a high-resolution camera in real time. The metrics called of accuracy, precision, sensitivity, and F1-Score were used to evaluate the performance of two different CNN models.

The use of Bosch data to detect faults in production lines has been implemented in three types of ML as Support Vector Classifier, Decision Tree Classifier and KNN Classifier [10].

Authors revised about the integration of ML algorithms to establish the quality of parts, saving time, energy and resources, and avoiding waste. Between 1995 and 2020, there is a report that 34% of publications related to production control applied an artificial neural network. In sequence, 14% was used the genetic algorithm, 13% was applied the Support Vector Machine (SVM), 10% was used the Random Forest, 7% was applied logistic regression and 5% was controlled the processes with Naïve Bayes [11].

A review of the last ten years of the challenges faced when applying AI - Artificial Intelligence in manufacturing processes is carried out, in addition to reviewing the main techniques [12]. One such challenge considered is data confidentiality where data can be revealed directly during transmission and processing.

To detect defects in parts in production, an intelligent system based on deep learning was developed, designed, and tested. The designed artificial intelligence system aims to learn which process parameters can make the product out of specification [13]. This knowledge means that the manual configuration is due to be replaced, since the process parameters are automatically adjusted.

It is to predict the quality in the milling process six trained and tested machine learning models were applied as Support Vector Machine (SVM), k-Nearest

Neighbors (KNN), Ridge Regression classifier (RidgeRe), Gaussian Naive Bayes classifier (GNB), Decision Tree (DT), Multi-Layer Perceptron (MLP) [14]. It is to evaluate the performance of the models, four equations were used as accuracy, precision, recall, and specificity. All these equations are described with parameters TP, TN, FP and FN being TP the true-positive predictions (true “OK” predictions), TN the true-negative predictions (true “NOK” predictions), FP the false-positive predictions, FN as false-negative predictions.

Just as connecting rods are engine components, bearings are also part of engines. To evaluate the bearing faults three classification algorithms were trained and tested as KNN, Decision Tree and Random Forest. The Confusion matrix was shown for each classifier [15]. The five equations called precision, sensitivity, specificity, prediction, and F1-s containing as parameters TP, TF, FP, and FN were used to predict the quality of the ML techniques.

The identification of defective components during manufacturing in the aerospace industry using real-time deep learning was proposed in [16]. In this case, convolutional neural network (CNN). For the project, 1,550 images of aircraft wingtip assemblies were used, which were taken using a high-definition camera. Of these images, 80% were used for training dataset and the dataset remaining 20% for the test dataset. The validation of the method was given applying equations of accuracy, precision and F1 score. All equations involving the four parameters: TP, TF, FP, and FN.

It to improve the prediction of production must understand the algorithmic behavior and results of machine learning methods [17]. Then, it was shown how mathematical techniques may be applied to machine learning methods to investigate the algorithmic behavior of some methods and therefore help improve their performance. The work shown calculations involving sums and integrals and, therefore, it was needed to have a more advanced mathematical knowledge to read it.

To predict possible valve failures, it was applied ML models such as linear regression and K-Means. Failures caused by valves in an industrial process, can cause line stoppages due to unforeseen breaks [18]. So, in addition to not causing these stops, ML techniques can also improve the operator's routine and reduce maintenance costs. Accuracy of machine learning methods are also used to verify cyber-attacks. A proposal to improve cybersecurity is the use of three different types of machine learning and for each one parameter that include accuracy are calculated [19].

To estimate the possibility of a cancer diagnosis in a person, machine learning techniques were used. In this way, five Random Forest methods Classifier, Support

Vector Machine, K-Nearest Neighbor (KNN), K-Means Clustering, Decision Tree Algorithm were used in the biomedical field and the precision of each method was calculated. The confusion matrix used in this work also follows the same model as the work proposed in this paper [20].

3. Theory and Calculation

Consider that for a better understanding of the machine learning models used in this work, it would be advisable to review the concepts of each of the seven models as Naïve Bayes, K-Nearest Neighbors (k-NN), Decision Tree, Support Vector Clustering (SVC), Logistic Regression and Neural Network. In this way, one by one is briefly described.

3.1 Naïve Bayes (Gaussian NB)

When it is assumed that all characteristics are independent given the class attribute, the Naïve Bayes classifier can be used, as shown in Figure 2 [21].

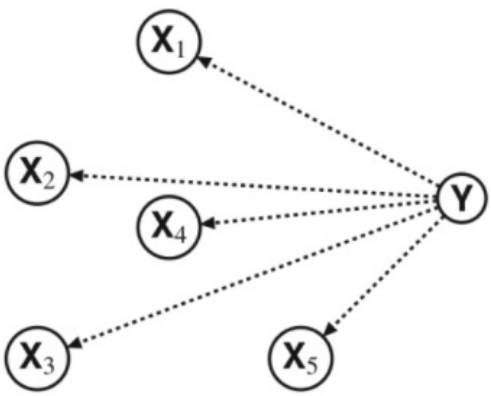

Figure 2. Topography of a Naive Bayes classifier where each feature Xi (i = 1, 2,..., 5) only depends on the class attribute. Y represents the class node and X represents the feature node [pp. 21].

It is necessary to understand Bayes' theorem to understand how the Naïve Bayes classifier works [22].
Mathematically writing, the joint probability between an event A and an event B to occur is given by:

$$P(A \cap B) = P(B)(A/B) \quad (1)$$

Or:

$$P(B \cap A) = P(A)(B/A) \tag{2}$$

Making the equations equal:

$$P(B)(A/B) = P(A)(B/A) \tag{3}$$

Isolating a term P(A/B):

$$P(A/B) = \frac{P(A)P(B/A)}{P(B)} \tag{4}$$

One way to get P(B) is given by:

$$P(B) = P(A \cap B) + P(A^C \cap B) \tag{5}$$

Onde: A^C is the complementary probability of A.

Replacing eq. (2) and (5):

$$P(B) = P(A)P(B/A) + P(A^C)P(B/A) \tag{6}$$

Replacing eq. (5) in (4), we have Bayes' theorem written as follows:

$$P(A/B) = \frac{P(A)(B/A)}{P(A)P(B/A)+P(A^C)P(B/A^C)} \tag{7}$$

If $P(A_j)$ and $P(B|A_j)$ are given, then P(B) is given by:

$$P(B) = \sum_j P(B/A_j)P(A_j) \tag{8}$$

Them Bayes' theorem written as follows:

$$P(A_j/B) = \frac{P(B/A_j)P(A_j)}{\sum_j P(B/A_j)P(A_j)} \tag{9}$$

An example of application of the Naive Bayes method where the default of short-term loans for a Tunisian commercial bank was predicted can be read in [23].

The Naive Bayes classifier was used on sentences containing these ambiguous words to remove ambiguity between sentences translated using two languages [24].

3.2 K-Nearest Neighbors (k-NN)

A famous classification technique of machine learning is called k-nearest neighbors (k-NN) [25].

It is one of the simplest predictive models that exists only based on the distance of the close neighbors, without taking into account any other type of information [26].

A model to improve the k-NN technique was applied in [27]. The k-NN was used together with the genetic algorithm to improve a ranking.

The k-NN method was applied to forecast fuel consumption to define an adequate energy budget, since there is a lot of expenditure to purchase fuel for agricultural machinery on farms [28].

3.3 Decision Tree

Decision trees (Figures 3 and 4) can be used to solve classification and regression problems. The advantages of using a decision tree are the use of less data and less use of computational resources [25]. These benefits left the most popular decision tree. The outputs for classification decision trees are categorical and regression variables are numerical outputs [26].

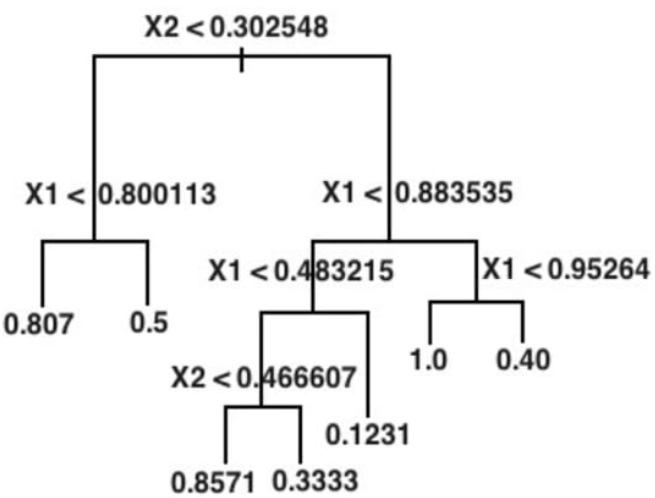

Figure 3. Classification tree: an example of categorical outcomes [25, pp. 102].

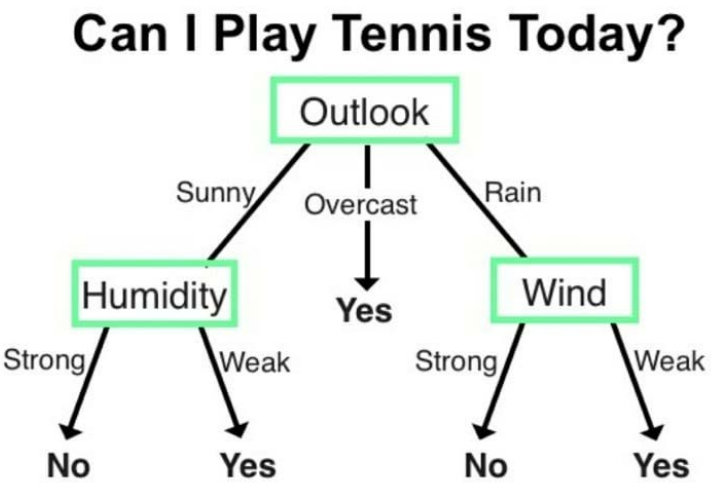

Figure 4. Regression tree: an example of numeric outcomes [25, pp. 102].

3.4 Random Forest

A problem that can be observed in decision trees is the possibility of overfitting training data [26]. A problem that can be observed in decision trees is the possibility of overfitting training data. To solve this problem, the random forest method can be used, which consists of obtaining several individual decision trees that depend on random characteristics. Each tree is different from the other.

3.5 Support Vector Clustering - SVC

The Support Vector Clustering (SVC) is a technique based on Support Vector Machine (SVM) with optimizing of the classification by a quadratic function [29].
In Figure 5 it can see data distributed in different ways. In two of them (a and c) the SVM can be used to build a data separation line. In case b, this is not possible. A quadratic function, rather than a linear one, would be more suitable in this case, so using SVC would be more interesting.

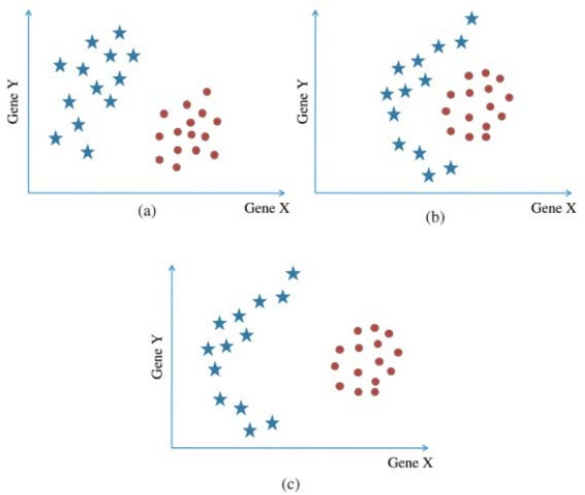

Figure 5. Examples of data where there is linear separation (a and c) and where linear separation cannot be used (b) [30, pp.140].

In SVM a hyperplane is built to group the data and in SVC a hypersphere with the smallest radius is built [31]. Support vectors are called the data that fit on the surface of the hypersphere. It is these data that determine the cluster's contour region (Figure 6).

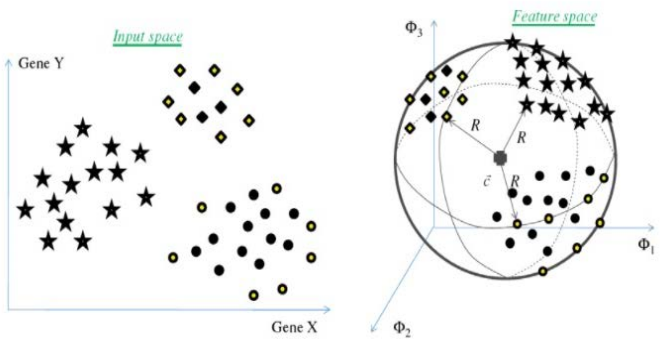

Figure 6. Example of a hypersphere of smaller radius that encompasses all data [30, pp. 138].

3.6 Logistic Regression

The Logistic Regression is a technique used to predict, in priority, qualitative predictions. [25]. Generally, two discrete classes are to be predicted, such as pregnant and non-pregnant.

The Logistic Regression is defined as a linear classifier (Figure 7) that calculates the probability of samples for two different classes [32].

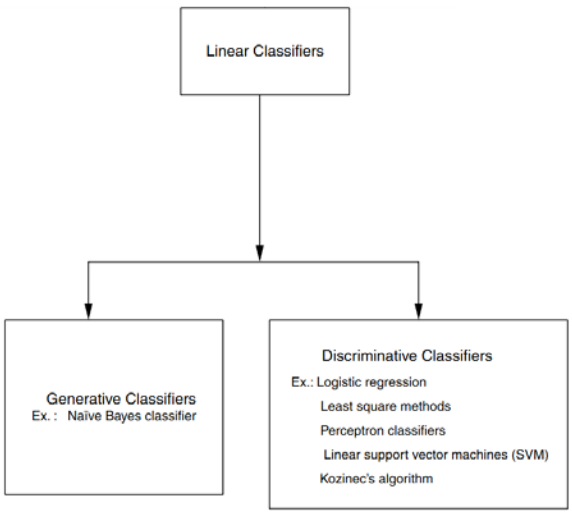

Figure 7. Logistic Regression is part of the linear classifiers [33, p.31].

Binary classification can be performed by applying a sigmoid function where any number is transformed into values between 0 and 1 (Figure 8). When applying a limit of 0.5, values below this value are classified as 0 and above are classified as 1. That is, class A or class B (Figure 9).

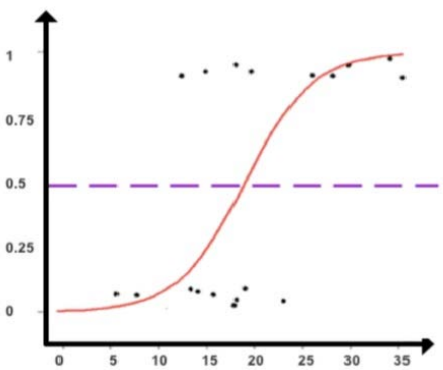

Figure 8. An example of using a sigmoidal function for binary classification [25, pp.64].

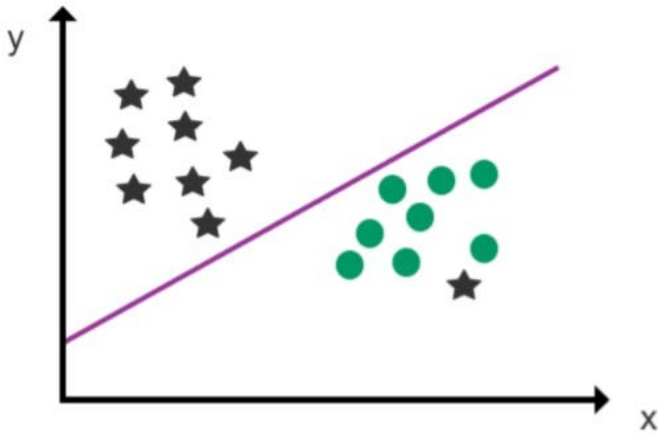

Figure 9. Use of the Logistic Regression technique to classify data [25, pp.66].

The Figure 10 shows the separation lines generated by two models: logistic regression and SVM [34]. In this case, the sepal and petal size data refer to a collection of flowers of the species Iris Setosa and Iris Versicolor. From what can be seen, logistic regression separates the binary classes through a separation line.

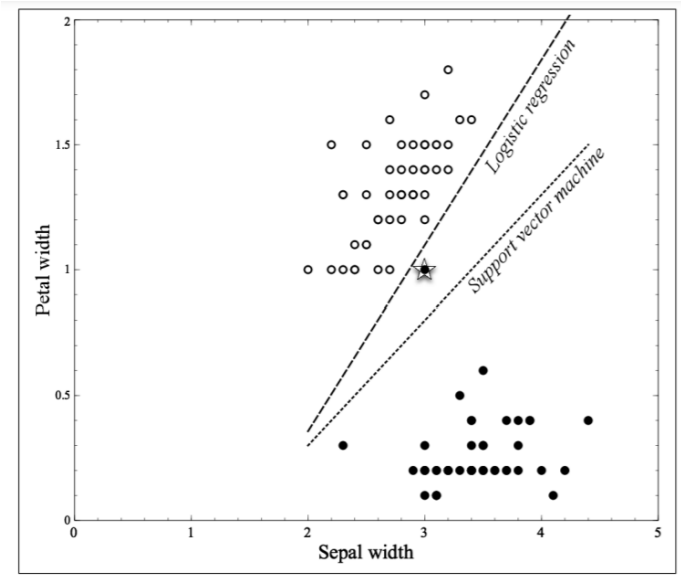

Figure 10. Using logistic regression and SVM to separate binary classes [34, p.89].

For classification in a scenario containing more classes, the multinomial logistic regression [25].

3.7 Neural Network

A network built with multiple neurons and one layer in the late 1950s was called a perceptron by Rosenblatt (Figure 11) [35], [36].

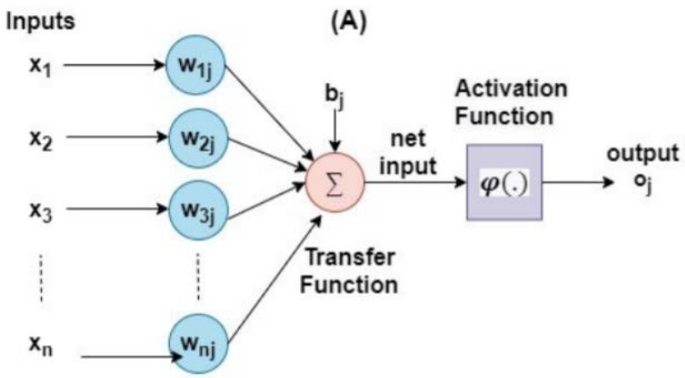

Figure 11. Original perceptron model proposed by Rosenblatt with one layer [35].

According to Figure 12 which represents a perceptron with four layers, going from the input layer (k = 0) to the output layer (k = 3). Intermediate layers (k = 1 and k = 2) are called hidden layers.

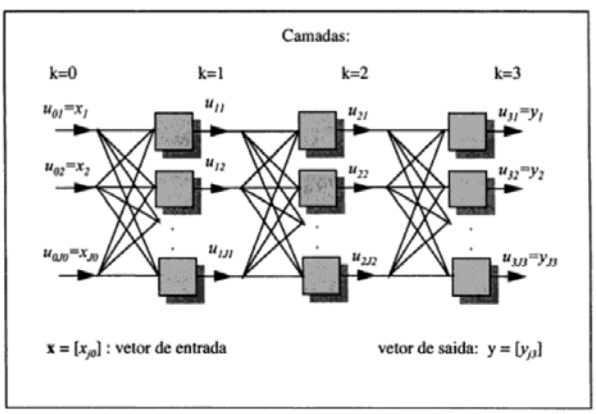

Figure 12. Four-layer neural network (multilayer perceptron) [36, p. 40].

It is described a case study using neural networks to calculate the energy consumed by a milling tool in a milling process [37]. Although there are several variables in the process, only three variables were chosen: feed rate, spindle speed and depth of cut. Although there are many types of neural networks, the authors opted for the multilayer perceptron (MLP) whose model is shown in the Figure 13.

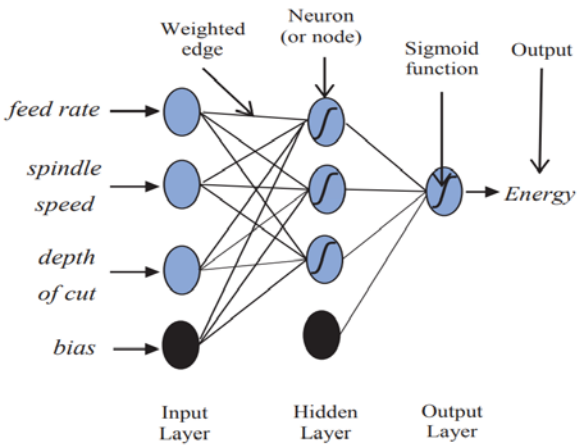

Figure 13. Example of MLP used to calculate energy consumption used in a milling tool [37, pp.1429].

In neural model according to Figure 13 there are three layers: input layer, hidden layer, and output layer. The values of the three parameters enter the input layer. In the output layer, represented by a neuron, energy prediction is performed.

To inspect online for surface defects in tiles during the manufacturing process used IR images and artificial neural networks, with defective and non-defective tiles [38].

3.8 Confusion Matrix

An accuracy of machine learning models can be evaluated through of the use of a matrix called a Confusion Matrix [39].

The number of classes indicates the size of the confusion matrix M, for example, it is a problem of separating two classes as in this work, a matrix is 2 x 2 as defined in Figure 14 a.

If the number of classes is C, then the matrix size will be C x C as shown in figure 14 b.

Analyzing a binary separation (1 and -1 classes), the matrix element M11 is called true-positive (TP), M12 element is called false-negative (FN), M21 element is called false-positive element (FP), and M22 element is called true-negative (TN).

It is M12 element referred to the number of ones that are of class 1 but, which were classified as -1.

Element M21 shows the number of elements that are -1 but classified as 1.

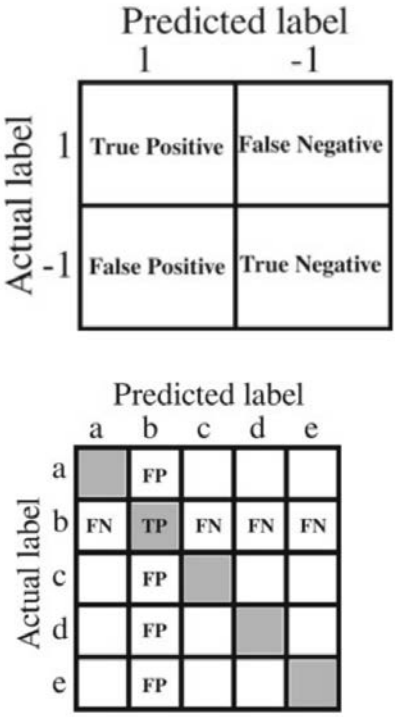

Figure 14. a) Confusion matrix for problems with binary classification and b) for problems with multiclass [39, pp.108 - 109].

The equation that calculates the accuracy is given by:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{10}$$

The machine learning model would be perfect if there were only diagonal elements, that is, FP = FN = 0.

4. Procedure

Two scripts in Python were written to carry out this work. The first one consists of simulating the data of the internal and external diameters of the connecting rod, where there would be one thousand simulated points (Figure 15) as input data and five hundred test points (Figure 16).

For the inner diameter, the program simulated one thousand random values between the values of 13.95 mm and 14.05 mm as input data and five hundred values as test

data.

For the outer diameter were simulated one thousand input values and the five hundred test data ranged between 21.95 mm and 22.05 mm.

The Python random number function was used to generate these diameters.

The simulations of diameters of connecting rods that satisfy the design specification are shown in Figures 15.

The simulated inner and outer diameter values were put together in two different columns, each containing one thousand or five hundred points, so that they can be plotted on a graph to verify which of them are in or out of a possible specification. Such chosen specification is a circumference of radius r of 0.04 mm and center 14.22 mm., that is, within this circumference the connecting rods would continue to be produced. If the values are outside the circumference contour, the connecting rods would be scrapped or could be machined again so that they can return the production line, avoiding material waste.

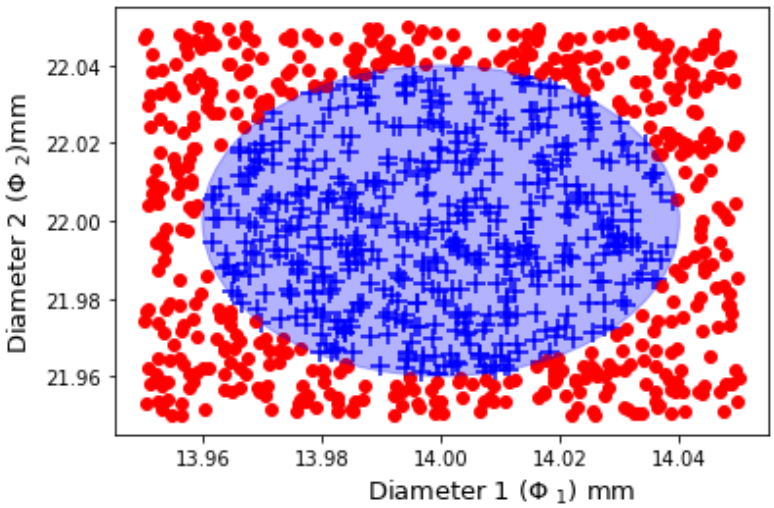

Figure 15. One thousand dice are used for training. Points blue marked with "+", inside the circle are within the values specified by the customer.

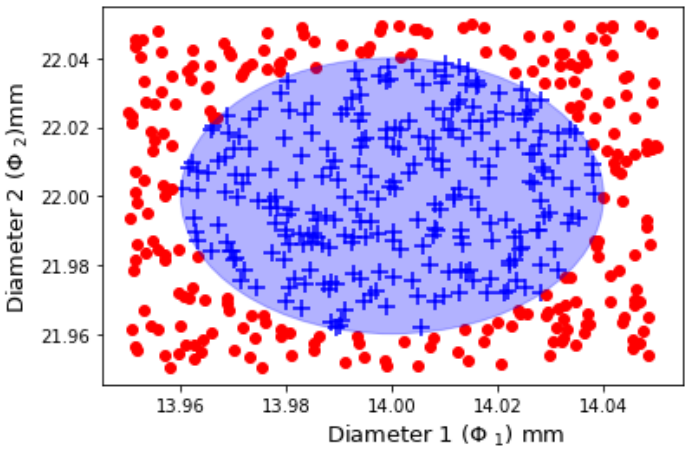

Figure 16. Five hundred test data were used to make the predictions. Points blue marked with "+", inside the circle, are within the values specified by the customer.

The second script consisted of reading the data from the first script and these data were training input data (one thousand pieces of data for each diameter) and test input data (five hundred pieces of data for each diameter) to each of the seven machine models learning.

The test size, which is parameter, is a number between 0 and 1, is a number that indicates the number of data used for testing within the training input values. The value used was 0.35, which means 35% of the thousand input data, corresponding to the value of 350 data. Remarkably high values of test size may indicate that there is not much data for training. There is no ideal test size number, and this number is chosen from according to the programmer's experience.

There are Python libraries that already have the models of each ML studied here, just the programmer already has experience with the parameters that are requested in the input.

For example, to import one of these libraries into logistic regression just write in the script: "from sklearn.linear_model import LogisticRegression". For the neural networks, the tensorflow library was imported.

Consider used parameters in each ML method:

• Decision Tree: criterion= entropy and random_state=0,

• Logistic Regression, solver= lbfgs and random_state=0,

• k-NN, n_neighbors=8, metric=minkowski and p=5,

• Random Forest: n_estimators=17, criterion= entropy and random_state=0,

• Gaussian NB: no option was chosen,

• SVC: kernel= rbf and random state= 1,
• Neural network: three layers being eight input neurons, four neurons in the hidden layer, and one neuron in the output layer. The number of epochs inserted that defines the number times that the learning algorithm worked was 600. The optimizer used was RMSprop, loss was mean squared error and, metrics was accuracy.

All this information above is inserted in a code of the standard script of the Python language. For each ML, a line of code is written including all of them.

To calculate the accuracy and the confusion matrix, two functions were created, one of which was for six types of ML used, except neural networks and the other, specifically for neural networks. Neural networks have the number of epochs as an input parameter and therefore, it was necessary to build a separate function from the other methods.

The output results obtained would be accuracy and confusion matrix where this matrix informs the true-positive (TP), false-negative (FN), false-positive element (FP), and true-negative (TN) values. With these values one can easily calculate the accuracy of each method. The diagonal of the matrix represents the TP and TN values.

5. Results and Discussion

Considering that the thousand training data and the five hundred test data, which refer to the diameters, are common data for the seven different types of machine learning used in this work, then, the Python program returns the confusion matrix (Table 1) for each method.

The confusion matrix for each machine learning model is shown in Table 1.

Table 1. Confusion matrix obtained for each classifier used in training.

Machine Learning	Confusion Matrix
Decision Tree	[[234 6] [15 245]]
Logistic Regression	[[114 126] [[119 141]]
k-NN	[[234 6] [15 245]]
Random Forest	[[228 12] [11 249]]
Gaussian NB	[[240 0] [47 213]]
SVC	[[237 3] [13 247]]

Neural Network	[[235 5] [10 250]]

The values for each accuracy shown in Figure 17 can be calculated using eq. (10) where $TP + TN + FP + FN = 500$ for all cases (total number of test data). The sum of diagonal elements $TP + TN$ results in the numerator value of that equation.

Decision Tree:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} = \frac{234+245}{500} \approx 0.96 \quad (11)$$

Logistic Regression:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} = \frac{114+141}{500} \approx 0.51 \quad (12)$$

K- NN:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} = \frac{234+245}{500} \approx 0.96 \quad (13)$$

Random Forest:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} = \frac{228+249}{500} \approx 0.95 \quad (14)$$

Gaussian NB:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} = \frac{240+213}{500} \approx 0.91 \quad (15)$$

SVC:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} = \frac{237+247}{500} \approx 0.97 \quad (16)$$

Neural Network:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} = \frac{235+250}{500} \approx 0.97 \quad (17)$$

All these calculated accuracy values were plotted to find out the best methods (figure 17).

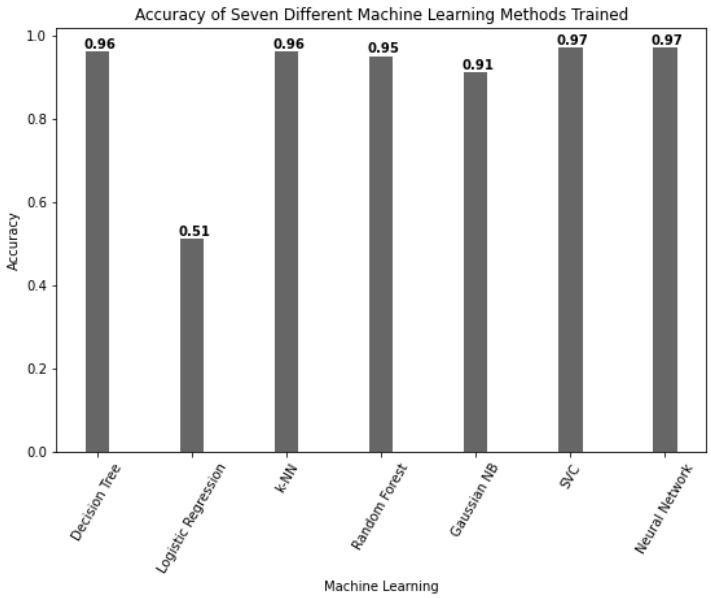

Figure 17. Accuracy obtained for each classifier used in training.

According to Figure 17, only the logistic regression presented a value considerably lower than the other methods. The explanation for this is explained that logistic regression is defined as a linear classifier and the data used cannot be separated linearly, as they have a circular symmetry.

For the other six types of ML, values were greater than 90%, reaching 97% for SVC and neural networks, which shows that good experience can be had with these methods on the production line. Logically, one can take more data to improve training for greater accuracy or even change the training input parameters. In the case of neural networks, for example, an independent study could be carried out to study the number of hidden layers that could be used for a better performance of the system. Of course, you can also modify the optimizer, loss, and metric parameters. The optimizer parameter adjusts parameters to minimize loss and the loss parameter measures predicted output of a model and the actual output.

However, it is worth emphasizing that there is a time in the connecting rod manufacturing line, and a very dense and complex neural network can delay production. Therefore, it is necessary to balance the accuracy value with the production time, since even though accuracy is achieved with high significant values, a high system response time is not desired to identify whether a connecting rod is

scrap or not.

6. Conclusion and Future Scope

To understand the accuracy of the seven machine learning methods used in this work, training data and test data are the common input parameters for each method. The Figure 15 shows the thousand simulated data referring to the variation of the connecting rod diameter and which are the input data for each method.

Figure 16 shows the data used for the test, in which five hundred different diameter values were simulated.

All programs were run in the Python programming language.

As the precision equation to be calculated was informed in eq. 10 and given that the Python simulations report the confusion matrix shown in Table 1, then according to Figure 17, the best techniques used were SVC and Neural Network, both with 0.97 accuracy. Next, appear K-Neighbors and Decision Tree, both with 0.96 accuracy.

The accuracy evaluated for Random Forest was 0.95. The accuracy of Gaussian NB is 0.91.

The only machine learning technique far below the value of all others is Logistic Regression.

Thus, the value for this classifier was lower than the other values.

Assuming that data with circular symmetry in a production environment, six machine learning techniques can be used with precision above 0.9. Obviously, calculation time is also important as well as accuracy.

In a perspective for the future, it would be to apply this method in a production line of a metallurgical company that manufactures connecting rods to verify the application this model directly in the production line and compare whether the connecting rods are within or outside the diameter parameters specified by the design created by the product engineering department.

Conflicts of Interest

It has no conflict of interest.

References

[1] P. R. Wani, "Connecting Rod," *Design and Development of Heavy-Duty Diesel Engines: A Handbook". Springer*, pp. **467-508**, 2020. https://doi.org/10.1007/978-981-15-0970-4

[2] A. Muhammad, M. A. H. Ali, I. H. Shanono, "Design Optimization of a Diesel Connecting Rod," *Materials Today: Proceedings*, Vol.**22**, Issue **4**, pp.**1600-**

1609, **2020**. https://doi.org/10.1016/j.matpr.2020.02.122

[3] M. Seyedzavvar. and M. Seyedzavvar, "Design of High Duty Diesel Engine Connecting Rod Based on Finite Element Analysis," *Journal of the Brazilian Society of Mechanical Sciences and Engineering*. Springer, Vol. **40**, Issue **59**, **2018**.

[4] M. T. Alam, A. Thakur, V. Kumar PS, S. Ghadei, "Fatigue Failure Analysis of Diesel Engine Connecting Rod," *SAE Technical Paper* 2018-28-0067, **2018.** https://doi.org/10.4271/2018-28-0067

[5] A. Muhammad, I. H. Shanon, "Static Analysis and Optimization of a Connecting Rod," *International Journal of Engineering Technology and Sciences (IJETS)*, Vol.**6**, **2019**. ISSN: 2462-1269.

[6] G. V. Shinde, V. D. Yadav, V. M. Jadamar, G. S. Jadhav, S. M. Sawant, "Stress Analysis of Connecting Rod using FEA," ICRAME2015, **2015**.

[7] L. Witek, P. Zelek, "Stress and Failure Analysis of the Connecting Rod of Diesel Engine," *Engineering Failure Analysis*. Elsevier. ScienceDirect. Vol. **97**, **pp**.374-382, **2019.**

[8] E. Lohith, K. S. S. Aradhya, N. G. S. Udupa, "Vibration Study of a Locomotive Engine Connecting Rod Using Consistent and Lumped Mass FEM Formulation Approaches," 59th Congress of ISTAM, 2014.

[9] E. Westphal, H. Seitz, "A machine learning method for defect detection and visualization in selective laser sintering based on convolutional neural networks," *Additive Manufacturing*, Vol. **41**, **2021**. https://doi.org/10.1016/j.addma.2021.101965

[10] S. C. Kong, M. Brungs, M. Nitiruksakul, B. Ennor, "Predicting production line performance with machine learning," *Medium*, **2022**.

[11] S. Kumar, T. Gopi, N. Harikeerthana, M. Kumar Gupta, V. Gaur, G. M. Krolczyk, C. Wu, "Machine learning techniques in additive manufacturing: a state-of-the-art review on design, processes and production control," *Journal of Intelligent Manufacturing*, Vol. **34**, pp. **21–55**, **2023**. https://doi.org/10.1007/s10845-022-02029-5

[12] J. Xu, M. Kovatsch, D. Mattern, F. Mazza, M. Harasic, A. Paschke, S. Lucia, "A Review on AI for Smart Manufacturing: Deep Learning Challenges and Solutions", *Applied Sciences*, Vol. **12**, Issue **16**, **2022**. https://doi.org/10.3390/app12168239

[13] N. M. Hassan, A. Hamdan, F. Shahin, R. Abdelmaksoud, T. Bitar, "An artificial intelligent manufacturing process for high-quality low-cost production," *International Journal of Quality & Reliability Management*, Vol.**40**, No **7**, pp. **1777-1794**, **2022**. https://doi.org/10.1108/IJQRM-07-2022-0204

[14] A. Fertig, M. Weigold, Y. Chen, "Machine Learning based quality prediction for milling processes using internal machine tool data," *Advances in Industrial and Manufacturing Engineering*, Vol.**4**, **2022**. https://doi.org/10.1016/j.aime.2022.100074

[15] R. Nishat Toma, A. Prosvirin, J. Kim, "Bearing Fault Diagnosis of Induction Motors Using a Genetic Algorithm and Machine Learning Classifiers," *Sensors*,

Vol.**20**, Issue **7**, **2020**. https://doi.org/10.3390/s20071884

[16] I. Shafi, M. F. Mazhar, A. Fatima, R. M. Alvarez, Y. Miró, J. C. M. Espinosa, I. Ashraf, "Deep Learning-Based Real Time Defect Detection for Optimization of Aircraft Manufacturing and Control Performance," *Drones 2023*, Vol. **7**, Issue **1**, **2023**. https://doi.org/10.3390/drones7010031

[17] M. Rom, M. Brockmann, M. Herty, E. Iacomini, "Machine learning tools in production engineering," *The International Journal of Advanced Manufacturing Technology*, Vol**.121**, pp. **4793–4804**, **2022**. https://doi.org/10.1007/s00170-022-09591-5

[18] M. C. Pinto, "Sistema de Manutenção Preditiva de Falhas em Válvulas em um Processo Industrial Utilizando Inteligência Artificial," Final paper, Universidade Federalde Uberlândia – UFU, Faculdade de Engenharia Elétrica, Uberlândia, **2021.**

[19] K. A. Okewale, I. R. Idowu, B. S. Alobalorun, F. A. Alabi, "Effective Machine Learning Classifiers for Intrusion Detection in Computer Network," *International Journal of Scientific Research in Computer Science and Engineering*, Vol**.11**, Issue.**2**, pp.**14-22**, **2023**.

[20] S. Garag, A. S. Nandeppanavar, M. Kudari, "Machine Learning Approaches for Prediction of various Cancer types," *International Journal of Scientific Research in Computer Science and Engineering*, Vol.**10**, Issue.**6**, pp.**01-08**, **2022**. https://doi.org/10.26438/ijsrcse/v10i6.18.

[21] C. Wan, "Hierarchical Feature Selection for Knowledge Discovery: Application of Data Mining to the Biology og Ageing," Springer, pp. **21**,**2019**. https://doi.org/10.1007/978-3-319-97919-9

[22] V. A. O. Zevarex, C. H. S. Santos, "Studies of the Implications of the Bayes Theorem in Natural Computing," *Revista Brasileira de Iniciação Científica*, **2020**.

[23] A. Krichene, "Using a naive Bayesian classifier methodology for loan risk assessment: Evidence from a Tunisian commercial bank," *Journal of Economics, Finance and Administrative Science*, Vol.**22**, No.**42**. Lima, **2017**.

[24] D. Pinto, D. Vilariño, C. Balderas, M. Tovar, B. Beltrán, "Evaluating n–gram Models for a Bilingual Word Sense Disambiguation Task," *Computación y Sistemas*, Vol.**15**, No.**2**, pp. **209-222**, **2011.**

[25] O. Theobald, "Machine Learning. For Absolute Begineers," **2017**.

[26] J. Grus, "Data Science Do Zero: Noções Fundamentais com Python," *Starlin Alta Editora*, **2016**.

[27] N. Suguna, K. Thanushkodi, "An Improved k-Nearest Neighbor Classification Using Genetic Algorithm," *IJCSI International Journal of Computer Science Issues*, Vol.**7**, Issue **4**, No **2**, **2010**.

[28] N. M. N. Al-Dosary, S. A. Al-Hamed, A. M. Aboukarima, "K-Nearest Neighbors Method for Prediction of Fuel Consumption in Tractor-Chisel Plow Systems," *Scientific Papers, Agricultural Machinery Management*. Eng. agríc., Vol.**39**, No **6**, **2019**.

[29] R. Xu, D. C. Wunshch, "Clustering," *IEEE Pressing*. **2009**.

[30] A. Statnikov, C. F. Aliferis, D. P. Hardin, I. Guyon, "A Gentle Introduction to Support Vector Machines in Biomedicine: Theory and Methods," *World Scientific Publishing*, USA, **2011**.

[31] B. Lim, X. Li, J. B. Zhang, and D. Shi, "Feature Discretization by Support Vector Clustering," *Information Sciences 2007*, pp. **797-803, 2007**. https://doi.org/10.1142/9789812709677_0111

[32] H. Yuan, J. Geng, and F. Bian, "Geo-Spatial Knowledge and Intelligence: 4th International Conference on Geo-Informatics in Resource Management and Sustainable Ecosystem," *GRMSE 2016 Hong Kong*, China, Springer, pp. **31**, **2016**. https://doi.org/10.1007/978-981-13-0893-2

[33] K. S. Rao, and S. G. Koolagudi, "Emotion Recognition using Speech Features," Springer. **2013**. https://doi.org/10.1007/978-1-4614-5143-3

[34] T. Fawcett, and F. Provost, "Data Science for Business: What you Need to Know About Data Mining and Data-Analytic Thinking," Published by O'Reilly Media. **2013**.

[35] A. Choudhury, S. Shakya, S. Kaushik, and V. Dutt, "Evaluating Cascade Prediction via Different Embedding Techniques for Disease Mitigation," *Big Data in Psychiatry and Neurology*, Elsevier, **2021**. https://doi.org/10.1016/B978-0-12-822884-5

[36] Z. L. Kovács, "Redes Neurais Artificiais," *Livraria da Física*, São Paulo, **2006**.

[37] D. Lechevalier, S. Hudak, R. Ak. Y.T Lee, and S. Foufou, "A Neural Network Meta-Model and its Application for Manufacturing," *IEEE International Conference on Big Data (Big Data)*, pp. **1428-1435**, **2015**. https://doi.org/10.1109/BigData.2015.7363903

[38] R. M. Andrade, and A. C. Eduardo, "Methodology for Automatic Process of the Fired Ceramic Tile's Internal Defect Using IR Images and Artificial Neural Network," *Technical Papers. J. Braz. Soc. Mech. Sci. & Eng.*, Vol. **33**, No **1**, **2011**.

[39] H. H. Aghdam, E. J. Heravi, "Guide to Convolutional Neural Networks: A Practical Application to Traffic-Sign Detection and Classification," *Springer*, pp. **108-109**, **2017**. https://doi.org/10.1007/978-3-319-57550-6

AUTHOR PROFILE

Luciana C. M. F. Diogenes was born in the city of São Paulo, Brazil in 1980. She earned a bachelor's degree in physics at Unicamp (University of Campinas, Brazil) in 2001, master's in physics at Unicamp in 2004 and Ph.D. in Mechanical Engineering at Unicamp in 2008, where a year of the doctorate was held at the Faculty of Engineering of Porto in Portugal. She completed an MBA in Project Management in 2014 at FGV - Fundação Getúlio Vargas, Brazil.

She has worked as a project specialist in research and development in ThyssenKrupp at Campo Limpo Paulista, Brazil for 5.5 years. She also served as a university professor at UEMG - Minas Gerais State University - in the cities of Frutal and Ituiutaba, both in Brazil. In UEMG Frutal, she acted as research coordinator for a one-year term. Currently, the author is dedicated to works as Python Developer and is open to new cooperations.