XLI CILAMCE
XLI Ibero-Latin American Congress on Computational Methods in Engineering
16-19 November, 2020 FOZ DO IGUAÇU PARANÁ - BRAZIL

# Machine Learning models to predict nonlinear simulations of net section resistance in steel structures

Raí L. Barbosa[1], Francisco Evangelista Junior[1]

[1]*Dept. of Civil Engineering, University of Brasília*
*Campus Universitário Darcy Ribeiro, 70910-900, Distrito Federal, Brazil*
*railuz.barbosa@gmail.com, fejr.unb@gmail.com*

**Abstract.** This paper applies Machine Learning techniques to predict computer simulations of net section resistance in bolted cold-formed steel connections of steel structures. Support Vector Regression (SVR) and Gaussian Process Regression (GPR) techniques where chosen to be used for the analysis due to their good performances on high dimensional data problems. One of the goals is to construct an efficient machine learning model with minimal training to the uncertainty quantification of the net-section resistance. The algorithms were trained using data set from expensive nonlinear finite element simulations where the resistance depends on the cross-section geometry, connection eccentricity and connection length. The finite element simulations were considered nonlinear due to the elastoplastic behavior of the steel. SVR and GPR were then compared by using standardized statistics measures with different cross-validation strategies. The results showed that SVR had a slightly better performance. In addition to that, it was possible to identify the best covariance function of each technique for this specific problem.

**Keywords:** Support Vector Machine, Gaussian Process Regression, Machine Learning, bolted connection, structural failure.

## 1 Introduction

Gaussian Process Regression (GPR) and Support Vector Regression (SVR) are Machine Learning (ML) techniques that can be used as a regression tool in a variety of problems, and they are able to find the solutions for them by learning with examples. Based on a variety of cases with already known answers, they are capable of finding a solution for a new set of data.

Studies applying ML techniques in steel structures are recent and very limited. Lee et al. [1] proposed a new approach to predict mechanical characteristics of corroded steel strands. Based on FE models of corroded wires, the ultimate strength and strain of the steel strands are predicted using Monte Carlo simulations. Okyere et al. [2] used GPR in their studies to predict reservoir porosity and permeability of the southern basin of the South Yellow Sea. Pham et al. [3] applied a modified SVR to correlates the compressive strength of a high-performance concrete (HPC) and its components based on a database of 239 samples, where experimental results have shown that the model is promising for the problem. Gopalakrishnan and Kim [4] checked the performance of an SVR model to predict the dynamic modulus |E*| of a hot-mix asphalt based on eight inputs parameters. The model presented a high linear correlation between the predicted and observed values. These studies showed that GPR and SVR can be valuable tools for regression problems.

Fleitas et al. [5] modeled bolted cold-formed steel angles connections under tension using finite elements (FE) to obtain the net section resistance and study how connection efficiency is influenced by the connected length in the longitudinal and transverse direction and the eccentricity in $x$ and $y$ directions. The FE analysis generated a database that its appropriate for a different approach of predicting the connection resistance using a ML model.

The purpose of this study is to verify the applicability and performance of SVR and GPR in the prediction of the net section resistance of bolted cold-formed steel angles under tension. Some analysis was performed with these techniques in order to identify the covariance function that best fit this problem. In addition to that, to obtain

a good performance with minimal training, it was tested different sizes for the training and validation dataset in order to identify which proportion gives the best performance to quantify the uncertainty of the net-section resistance.

# 2 Background

## 2.1 Gaussian Process Regression

According to Rasmussen and Williams [6], Gaussian Process (GP) can be defined as a collection of random variables which have a joint Gaussian distribution and is fully described by its mean function *m(x)* and covariance function *k(x, x')*. The mean and covariance functions of a real process *f(x)* can be written as shown in eq. (1) and eq. (2):

$$m(x) = \mathrm{E}[f(x)] \tag{1}$$

$$k(x, x') = \mathrm{E}[(f(x) - m(x))(f(x') - m(x'))] \tag{2}$$

while the Gaussian Process is represented by eq. (3):

$$f(x) \sim GP(m(x), k(x, x')) \tag{3}$$

where *x* represents the input values.

According to Wang et al. [7], one of the advantages of GPR method is that its optimum hyperparameters can be estimated based on the maximum marginal likelihood method. However, Baraldi et al. [8] alerts that such optimization process can lead to an overfitting problem, and they assure that, due to the presence of multiple local maxima in the optimization function, the optimum point might not converge to the global maxima. Chen and Wang [9] said that, although the initial guess for the hyperparameters makes a difference in finding the best final parameters, it does not have a big impact in the model prediction.

## 2.2 Support Vector Regression

Awad and Khanna [10] stated that Support Vector Regression (SVR) is characterized by the use of a covariance function, VC control of the margin and the number of support vectors. According to Lin and Wang [15], the covariance function maps the input points into a high-dimensional feature space and finds a region for prediction that is limited by the separating hyperplane. The optimal hyperplane is influenced by a combination of few input points, called support vectors. They also affirm that the a SVR problem can be simplified as an ε-insensitive region around the function, called the ε-tube, where the solution is found by minimizing this region and finding the flattest tube that encompasses most of the training instances. For one-dimension example, a mathematical expression for the continuous-valued function is shown in eq. (4):

$$y = f(x) = < w, x > + b = \sum_{j=1}^{M} w_j x_j + b, y, b \in \mathbb{R}, x, w \in \mathbb{R}^M \tag{4}$$

where *w* is the weight vector, *b* is the bias term and M is the order of the polynomial to approximate a function. The <-,-> denotes the dot product in the input space.

Finding the flattest tube is seeking for a small *w*, which can be done by minimizing its norm. In a comprehensive manner, this can be done by controlling each free hyperparameters and finding the combinations that gives the best accuracy.

# 3 Methodology

## 3.1 Problem description and data

The database used for this paper was obtained from Fleitas et al. [5] study on net section resistance in bolted

connections on steel angles. A FE analysis was performed with the software ABAQUS and the geometric nonlinearity, the material nonlinearity and the contact between bolt, gusset plate and cold-formed angles were all introduced in the numerical model. The geometric parameters of the cold-formed bolted connection used as input for the training are shown in Figure 1. The FE analysis generated a database of 107 samples, and the inputs and output used in the models, with the range of their values, are shown at Table 1.

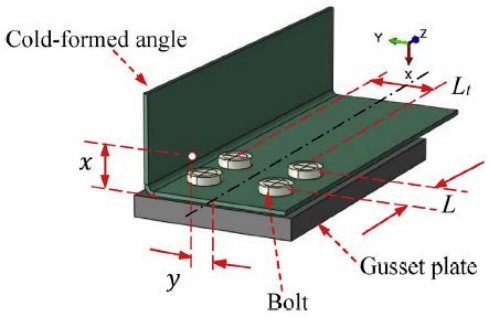



Figure 1 - Bolted connection and the geometric parameters used as input for training (Adapted from [5]).

Table 1 - Minimum and maximum for the features used in the models.

| Parameter | | Min | Max |
|---|---|---|---|
| Width of the angle connected leg (mm) | $b_c$ | 70 | 150 |
| Width of the not connected leg (mm) | $b_d$ | 50 | 130 |
| Distance from shear plane to the centroid of the cross-section (mm) | x | 7.4 | 43.7 |
| Distance from the centroid of the connection to the centroid of cross-section (mm) | y | 13.7 | 38.7 |
| Length of connection in the longitudinal direction (mm) | L | 33.87 | 76.2 |
| Length of connection in the transverse direction (mm) | $L_t$ | 30 | 60 |
| Resistance of the net section calculated using the FE analysis (kN) | $T_{FE}$ | 77.96 | 137.25 |

The bolt diameter, thickness of cold-formed angle, number of bolt lines and number of holes per bolt line are equal to 12.7 mm, 2.66 mm, 2 and 2 respectively. The Young's modulus, the yield strength, the ultimate tensile strength and strain hardening used was 210,000 MPa, 368 MPa, 502 MPa and 28.6% respectively.

### 3.2 Analysis

For the training and testing it was used the software MATLAB, which has built-in functions that performs regression analysis. Initially, it was tested which covariance functions would give optimal results for this problem. For GPR, five covariance functions were tested: rational quadratic, squared exponential, Matern 5/2, Metern 3/2 and exponential. While for SVR was tested four: gaussian, linear, quadratic and cubic functions. For all of them, the data was set to be standardized and, for an initial analysis, 20% of the data was used as holdout validation (HOV), which means that 80% of the data was used for training and 20% was used for testing.

According to Chen and Wang [9], the simplest way to evaluate the accuracy of the model is by using the root mean squared error, RMSE, which is defined in eq. (5):

$$RMSE = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(y'_i - y_i)^2} \tag{5}$$

where $y'_i$ and $y_i$ are the predicted and actual values, respectively.

Nevertheless, they affirm that this parameter can be influenced by the scale of the output values. For this reason, instead using RMSE, they suggest to use the standardized root mean square error, sRMSE, which is the RMSE divided by the standard deviation, $\sigma_y$, of $y_i$. Because of that, the parameter used to measure the performance of the models is the sRMSE along with the linear correlation coefficient, $R^2$.

The algorithm structure for determining the best training and validation dataset size can be outlined as follow: 1. The hyperparameters are defined using an optimization process; 2. A vector of several HOV's is defined so that

the code can calculate the sRMSE and $R^2$ for each one; 3. The training and validation of the algorithm are performed inside a loop, which means that, for a determined set of hyperparameters and a specific HOV, several training and validation are performed. The final numbers for each HOV are given as the mean of all the runs. After that, the process is repeated with a specific HOV and the algorithm is ready for new predictions.

In addition to holdout validation, a k-fold cross validation analysis was performed so that a comparison could be made between them. Since the database used in this paper is not too large, a k-fold cross validation could give more realistic results. In this strategy, the data is divided in $k$ subsets, where the algorithm starts using $(k - 1)$ subsets for training and one for validation. After that, the algorithm uses another subset for validation, and the remaining for training. By the end of the training, all subsets will have been used for training and validation. After the regression training, a Monte Carlo (MC) simulation was done to produce $1\times10^6$ data sample with similar distribution to the FE data. The simulation generated the predictions for the stress in order to perform the uncertainty quantification through the estimation of statistical features of the Monte Carlo sample such as probability density functions and the four statistical moments.

## 4 Results

For each covariance function analysed, the hyperparameters was selected by an optimization function. For the GPR functions, the covariance functions could be set as nonisotropic or isotropic. After performing 15 different tests with the initial HOV of 20%, the sRMSE were measured so that the norm, given by eq. (6), and the mean could be computed, as shown in Table 2. Based on these results, the covariance function selected for GPR and SVR was nonisotropic squared exponential and quadratic, respectively.

$$norm = \sqrt{\sum_{i=1}^{n} sRMSE^2} \tag{6}$$

where $n$ is the total amount of different tests.

Table 2 – Summary of the sRMSE tests for covariance function selection.

| **Covariance function** | **Norm** | **Mean** |
|---|---|---|
| *Nonisotropic Rational Quadratic* | 0.617 | 0.174 |
| *Isotropic Rational Quadratic* | 0.743 | 0.213 |
| *Nonisotropic Squared Exponential* | 0.471 | 0.138 |
| *Isotropic Squared Exponential* | 0.630 | 0.184 |
| *Nonisotropic Matern 5/2* | 0.784 | 0.221 |
| *Isotropic Matern 5/2* | 0.563 | 0.160 |
| *Nonisotropic Matern 3/2* | 0.494 | 0.137 |
| *Isotropic Matern 3/2* | 0.624 | 0.181 |
| *Nonisotropic Exponential* | 0.569 | 0.162 |
| *Isotropic Exponential* | 0.568 | 0.166 |
| *Gaussian* | 0.614 | 0.167 |
| *Linear* | 0.809 | 0.243 |
| *Quadratic* | 0.609 | 0.175 |
| *Cubic* | 0.681 | 0.195 |

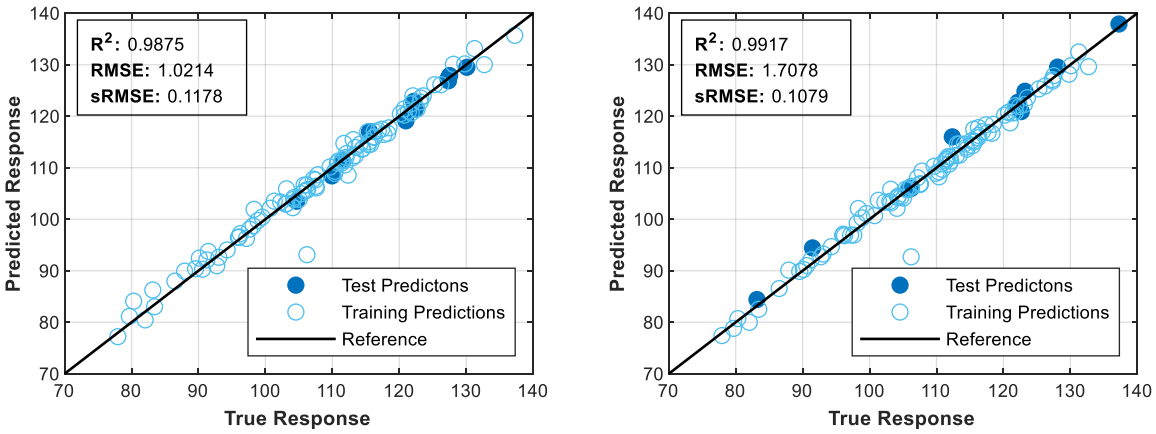
Table 3 shows the summary of the parameters used for selecting the HOV that will be used for prediction purpose. Six different tests were performed, where for tests 1, 2 and 3, the HOV varied from 10% to 50%, with an increment of 1%, and it was performed 100 repetitions of trainings and validations. For tests 4, 5 and 6, HOV varied from 10% to 20%, with 500 repetitions. From each test, it was possible to identify the HOV with the smallest sRMSE and its correspondent $R^2$.

From these results, a HOV of 11% was selected for both ML techniques to perform a one-time training and validation. Figure 2 shows the results from the training algorithm for the covariance functions and HOV selected. The figure compares the true response, obtained from FE analysis, and the predicted response, obtained from the ML model. The blue filled marks are the results from the validation dataset, which represents 11% of total sample. The marks with blue edges are the data used for the training. To obtain the predictions for the training dataset, the

algorithm was used after the training was done. The closest the marks are to the reference line, the better is the algorithm accuracy. It is possible to note that both had an overall good performance.

Table 3 – Summary of the tests for HOV selection.

| Test | Repetitions | HOV | sRMSE means | $R^2$ means |
|---|---|---|---|---|
| *SVR 1* | 100 | 17% | 0.1197 | 0.983 |
| *SVR 2* | 100 | 10% | 0.1231 | 0.982 |
| *SVR 3* | 100 | 11% | 0.1205 | 0.983 |
| *SVR 4* | 500 | 12% | 0.1271 | 0.980 |
| *SVR 5* | 500 | 11% | 0.1300 | 0.980 |
| *SVR 6* | 500 | 13% | 0.1288 | 0.980 |
| *GPR 1* | 100 | 12% | 0.1270 | 0.982 |
| *GPR 2* | 100 | 14% | 0.1303 | 0.979 |
| *GPR 3* | 100 | 11% | 0.1153 | 0.985 |
| *GPR 4* | 500 | 12% | 0.1302 | 0.980 |
| *GPR 5* | 500 | 10% | 0.1302 | 0.979 |
| *GPR 6* | 500 | 11% | 0.1282 | 0.980 |




a) GPR b) SVM

Figure 2 - True vs predicted response for GPR (a) and SVM (b) with a HOV of 11%.

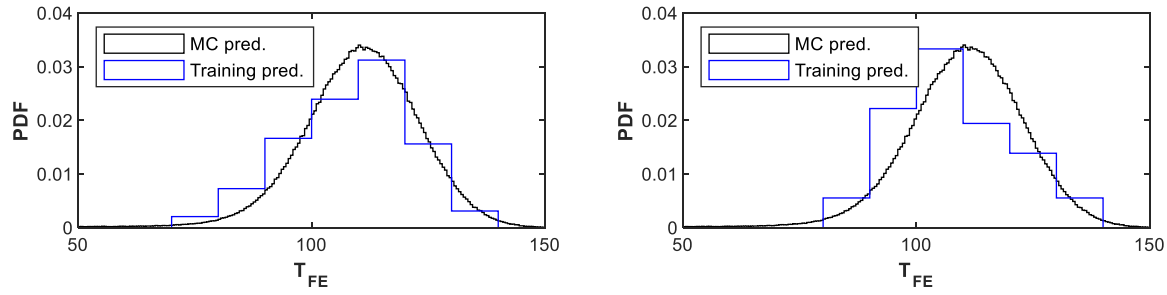
In order to check the performance of the data with a k-fold cross validation, k was set as 5 and three rounds of training and test was performed. The statistics results are shown at Table 4. The 5-fold cross validation model had a linear correlation close to the unity, which means that the model is able to capture the output variance.

Figure 3 shows histograms of the MC predictions, in black, and the FE response of the sample used for training the algorithm, in blue, for HOV's of 11% and 67%. The MC dataset contains a total number of $1\times10^6$ instances, while the training response dataset for HOV of 11% and 67% have 96 and 36 instances, respectively. These results show that the trained model was able to capture the PDF curve, from MC predictions, even for HOV of 67%. The PDFs curves are in very good agreement disregarding the HOV, while the histograms based on the training predictions, which use much reduced number of instances, show different shapes for the HOVs e for each ML method.

Table 5 illustrates some other statistics, such as the statistical moments (mean, standard deviation, skewness and kurtosis) of some data sample used in the present study. The first column shows the statistics of the FE response for the original dataset. The second and third columns show the statistics for the FE response of the dataset used as training sample for HOV of 11% and 67% respectively. And the fourth and fifth columns show the statistics for the predictions of the MC dataset using the algorithm trained with a HOV of 11% and 67% respectively. From the numbers in Table 5, SVR mean, standard deviation, skewness and kurtosis for the MC simulations are closer to the statistics of the training and testing sample than the ones for GPR model.

Table 4 - Statistics for 5-fold cross validation.

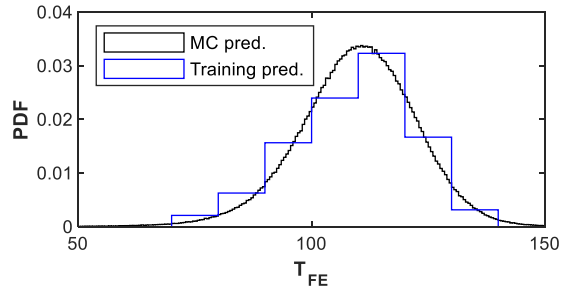
| | Test | $R^2$ mean | RMSE mean | sRMSE mean |
|---|---|---|---|---|
| **SVR** | First | 0.980 | 1.8558 | 0.1427 |
| | Second | 0.980 | 1.8453 | 0.1425 |
| | Third | 0.977 | 1.9506 | 0.1497 |
| **GPR** | First | 0.977 | 1.9547 | 0.1514 |
| | Second | 0.978 | 1.9324 | 0.1488 |
| | Third | 0.978 | 1.9080 | 0.1474 |



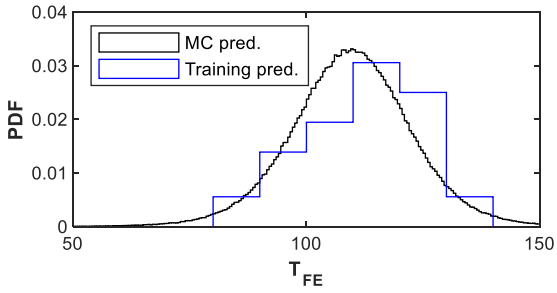

a) GPR with HOV of 11%.




b) GPR with HOV of 67%.




c) SVR with HOV of 11%




d) SVR with HOV of 67%

Figure 3 – PDF histogram comparing the GPR (a and b) and SVR (c and d) training predictions with HOV of 11% (a and c) and 67% (b and d) and predictions for the MC simulations.

Table 5 - Mean, standard deviation, skewness and kurtosis of the models.

| **GPR** | | | | | |
|---|---|---|---|---|---|
| | **Training and testing sample** | **Training sample** | | **MC Simulation Predictions** | |
| | | **HOV 11%** | **HOV 67%** | **HOV 11%** | **HOV 67%** |
| *Mean* | 109.2034 | 107.9716 | 107.5436 | 108.7305 | 108.7374 |
| *Standard Deviation* | 12.9311 | 12.8154 | 12.4789 | 19.312 | 19.2879 |
| *Skewness* | -0.3738 | -0.3061 | 0.2555 | -4.0062 | -4.0036 |
| *Kurtosis* | 2.6818 | 2.6947 | 2.9296 | 29.6105 | 29.6209 |
| **SVR** | | | | | |
| | **Training and testing sample** | **Training sample** | | **MC Simulation Predictions** | |
| | | **HOV 11%** | **HOV 67%** | **HOV 11%** | **HOV 67%** |
| *Mean* | 109.2034 | 108.6493 | 111.8792 | 109.7252 | 109.531 |
| *Standard Deviation* | 12.9311 | 12.4898 | 12.2722 | 12.6547 | 13.3423 |
| *Skewness* | -0.3738 | -0.4109 | -0.5566 | -0.3939 | -0.1378 |
| *Kurtosis* | 2.6818 | 2.7314 | 2.629 | 4.4778 | 4.6532 |

# 5 Conclusions

Although SVR and GPR numbers for sRMSE are very close, GPR presented the smallest and biggest values and SVR exhibited less variability. In this case, if it is not possible to do a large number of repetitions for the predictions, SVR would more appropriate to be used. Both of them showed to have a better performance when the HOV is set between 10% and 15%. Applying the HOV of 11% for SVR and GPR algorithms, with no repetition of training and validation, SVR returned a $R^2$ and a sRMSE of 0.9917 and 0.1079 respectively, while GPR returned 0.9875 and 0.1178, respectively. Both techniques presented a high linear correlation between the predicted and expected net-section resistance, $R^2$ close to the unity, which indicates that these techniques presented a good fit for this problem.

Comparing the k-fold cross validation with the holdout validation strategy, it is possible to note that the k-fold cross validation had less variability than the holdout validation. Therefore, the k-fold cross validation would be more reliable for smaller number of repetitions for the training and testing, since its $R^2$ still represents a good fit for the model. Comparing the histograms from Figure 3 and the statistics from Table 5, it is possible to note that the predictions of the MC simulations using SVR model are more similar to the data used in training and testing than the predictions made with GPR model. The PDFs curves are in very good agreement, disregarding the HOV, while the histograms based on the training predictions show different shapes for the HOVs for each ML method. This shows that a Monte Carlo simulation using either SVR or GPR is able to predict the probability density curve of the output variable (net-section resistance) even with an HOV of 67%.

Overall, the SVR and GPR algorithm showed to be efficient for the prediction of net-section resistance of bolted steel angles connections, with the SVR model performing slightly better. Despite the fact that these models had some variability of the results for different trainings, this problem can be easily solved increasing the number of trainings.

**Authorship statement.** The authors hereby confirm that they are the sole liable persons responsible for the authorship of this work, and that all material that has been herein included as part of the present paper is either the property (and authorship) of the authors, or has the permission of the owners to be included here.

# References

[1] J. Lee et al. “Probabilistic prediction of mechanical characteristics of corroded strands”. *Engineering Structures*, v. 203, 2020.
[2] S. A. Okyere et al. “Investigating the predictive performance of Gaussian Process Regression in evaluating reservoir porosity and permeability”. *Energies*, vol. 11, n. 12, 2018.
[3] A. Pham et al. “Predicting compressive strength of high-performance concrete using metaheuristic-optimized least squares support vector regression”. *Journal of Computing in Civil Engineering*, vol. 30, n. 3, 2016
[4] K. Gopalakrishnan and S. Kim. “Support Vector Machines approach to HMA stiffness prediction”. *Journal of Engineering Mechanics*, vol. 137, n. 2, pp. 138-146, 2011.
[5] I. Fleitas, J. Bonilla, L. M. Bezerra and E. Mirambell. “Net section resistance in bolted cold-formed steel angles under tension”. *Journal of Constructional Steel Research*, 2019.
[6] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for machine learning*. The MIT Press, 2006.
[7] F. Wang, J. Su and Z. Wang. “Prediction of subsidence of buildings as a result of earthquakes by Gaussian Process Regression”. *Chemistry and Technology of Fuels and Oils*, vol. 53, n. 5, 2017.
[8] P. Baraldi, F. Mangili and E. Zio. “A prognostics approach to nuclear component degradation modeling based on Gaussian Process Regression”. *Progress in Nuclear Energy*, vol. 78, pp. 141-154, 2015.
[9] Z. Chen and B. Wang. “How priors of initial hyperparameters affect Gaussian Process Regression models”. *Neurocomputing*, vol. 275, pp. 1702-1710, 2018.
[10] M. Awad and R. Khanna. *Efficient learning machine: theories, concepts, and applications for engineers and system designers*. Springer, 2015.
[11] C. Lin and S. Wang. “Fuzzy Support Vector Machines”. *IEEE Transactions on Neural Networks*, vol. 13, n. 2, 2002, pp. 464-471.