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

Concrete compressive strength prediction with machine learning

Pedro B. A. Moreira[1], Victor M. Silva[1]

[1]*Student, Dept. of Engineering, University Veiga de Almeida*
Avenida das Américas, 22631-004, Rio de Janeiro, Brazil
pedrobermoreira@gmail.com
[2]*Assistant Professor, Dept. of Engineering, IBMEC/RJ*
Avenida Armando Lombardi 940, 22640-000, Rio de Janeiro, Brazil
victor.silva@professores.ibmec.edu.br

Abstract. Compressive strength is the main characteristic of concrete. The correct prediction of this parameter means cost and time reduction. This work built predictive models for 6 different ages of concrete samples. A set of 1030 samples from previous studies was used, with 9 variables. Another 6 variables were added to represent the proportions of the main ingredients in each sample. The predictive models were developed in R language, using the Parallel Random Forest algorithm and repeated cross-validation technique to optimize the parameters. The results were compatible with other studies using the same data set. The most important model, 28 days old, obtained a root mean square error (RMSE) of 4.717. The 3-day model obtained the best result, RMSE of 3.310. The work showed that the compressive strength of concrete can be predicted. The choice of creating a model for each age allowed to get compatible results with the available data at each age. It was a promising alternative since good results were achieved by training with just one algorithm. This work facilitates exploration and new efforts to predict the compressive strength of concrete, it can be used as a baseline to predict with different algorithms or the combination of several.

Keywords: Concrete, Compressive Strength, Machine Learning, Prediction, Parallel Random Forest

1 Introduction

Compressive strength is the main characteristic of concrete, measured by tests of international standards that consist of breaking of specimens [1]. Measurement when the concrete is 28 days old is mandatory and represents the grade of the concrete. Knowing in advance the result for a given age, based on the proportions of its ingredients, is of great interest to concrete manufacturers, construction companies, and civil engineers.

The compressive strength is a nonlinear function of its ingredients and age, making it difficult to establish an analytical formula, although some formulas have already been proposed by Hasan and Kabir [2] and Kabir et al. [3]. However, most of the studies have built models including the age as a feature along with the ingredients, but due to the non linearity between the compressive strength and age, we have found the need of further investigation of models that separate the age and analyse only the ingredients, then specify for each age.

Therefore, the present study aims at building predictive models for the concrete compressive strength at different ages using only it's ingredients as features.

2 Related work

Yeh [4] demonstrated the possibility of using Artificial Neural Networks to predict the compressive strength of concrete, concluding that it is a more accurate method than regression models. In this study, more than 1000 concrete samples were collected from 17 different sources. This data set was later used in several studies about concrete, some of which are mentioned below.

Alshamiri et al. [5] proposed a new Regularized Extreme Learning Machine (RELM) method to train Artificial Neural Networks models to predict the compressive strenght. The results were compared with several known algorithms running on the same dataset, including individual and ensembles, and the proposed model had the best result by far.

Hameed and Khalid [6] compares Artificial Neural Network models with Multiple Linear Regression to predict the compressive strength force and have found that Artificial Neural Network models obtain much more accuracy than the Multiple Linear Regression.

In addition to these published studies, it is now very common to publish side projects on web pages. For easy access to this database and the growing interest in data science and machine learning, some unpublished studies using this same database include Modukuru [7], Raj [8], Abban [9] and Pierobon [10]. Overall, they all followed standard steps in the development of machine learning models, the first two using the scikit-learn package in python language developed by Pedregosa et al. [11] and both the latter used the caret package developed by Kuhn [12] in R language [13].

At the end of this work, in the discussion and conclusion section, the results found in this work are compared with all these related studies.

3 Materials and methods

3.1 Materials and reproducibility

The methodology was carried out using RStudio software [14], an integrated virtual environment for code development in R language [13]. Throughout the process, all code executed was documented in the same order as its execution and pushed to the github repository [15]. The repository contains an extended version of this paper, including all the code, required packages and versions. In order to guarantee reproducibility, whenever there was code that uses probabilistic operations, a seed was defined before its execution, ensuring results consistency when running on another machine.

3.2 Dataset

The data was downloaded from the University of California Irvine website [16]. In total there are 1030 rows with 9 columns. Each row represents a sample with the variables: compressive strength, age, and 7 ingredients (water, cement, fine aggregate, coarse aggregate, fly ash, blast furnace slag, and superplasticizers).

3.3 Data preparation

The related works used the data set in its entirety or performed a minimum of preparation. In a different way, in this work a specific step was dedicated just to clean the samples and prepare them for the next steps. The major steps executed in this section are listed below:

1. Duplicate samples were removed;
2. Samples aggregated and identified (with a new ID column) by the proportion settings of ingredients, independent of its age;
3. Ages of 90, 91, and 100 days were merged into a single 100-day category. This step was taken with the following development: first, plotting and analysing the boxplot grouped by age in Fig. 1 showed that samples of 90, 91 and 100 days have distinct concentrations of compressive strength values. Then a principal component analysis (PCA) of the ingredients were made in Fig. 2, showing that the ingredients combination of these ages are very distinct. As they are very close ages, it is reasonable that we can join these ages without any prejudice to the predictions. Finally, the 100-day mark was chosen because the analysis of samples that had data on at least five different ages showed that the compressive strength tends to increase through time. Therefore, an older age tends to provide more conservative results.
4. After joining the ages of 90, 91 and 100 days, the ages with a frequency less than 50 were removed, leaving only the ages of 3, 7, 14, 28, 56 and 100 days;
5. For samples with the same ID and the same age, but with different values of compressive strength, the data was combined into a single row for each combination of ID and age, containing an average of the compressive strengths;
6. Only the IDs that have data on the 28-day mark were kept;
7. Addition of 6 new continuous variables representing the proportions between the main ingredients that were used in the prediction models (water/cement, fine aggregate/cement, coarse aggregate/cement, fine aggregate/coarse aggregate, water/coarse aggregate and water/fine aggregate);
8. Addition of 2 new categorical variables used to visualize the distribution of the samples (concrete class and approximate mix).

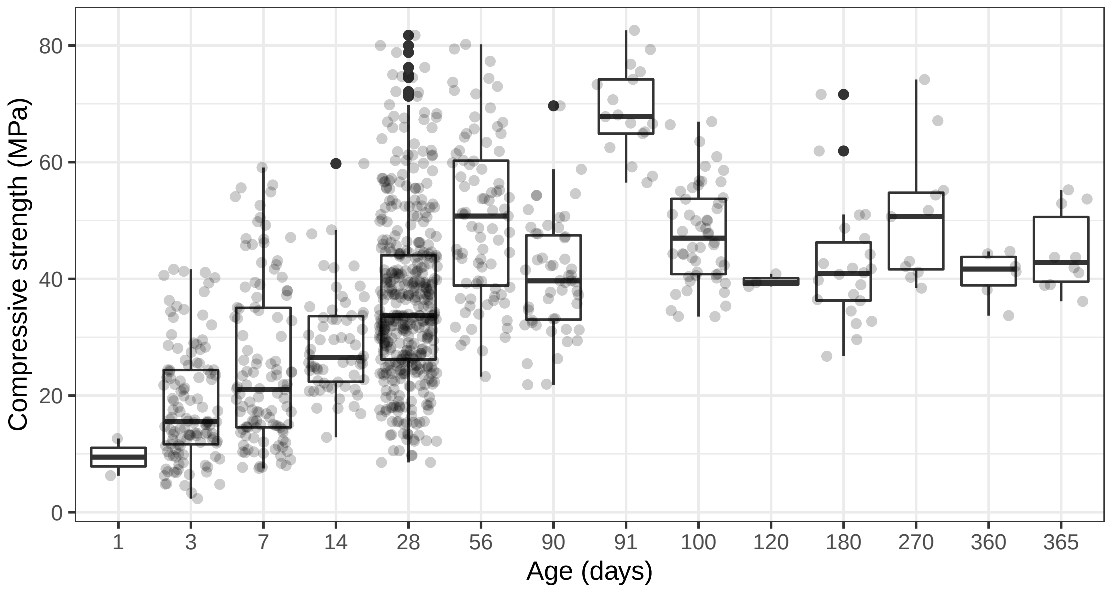

Figure 1. Boxplot - compressive strength grouped by age

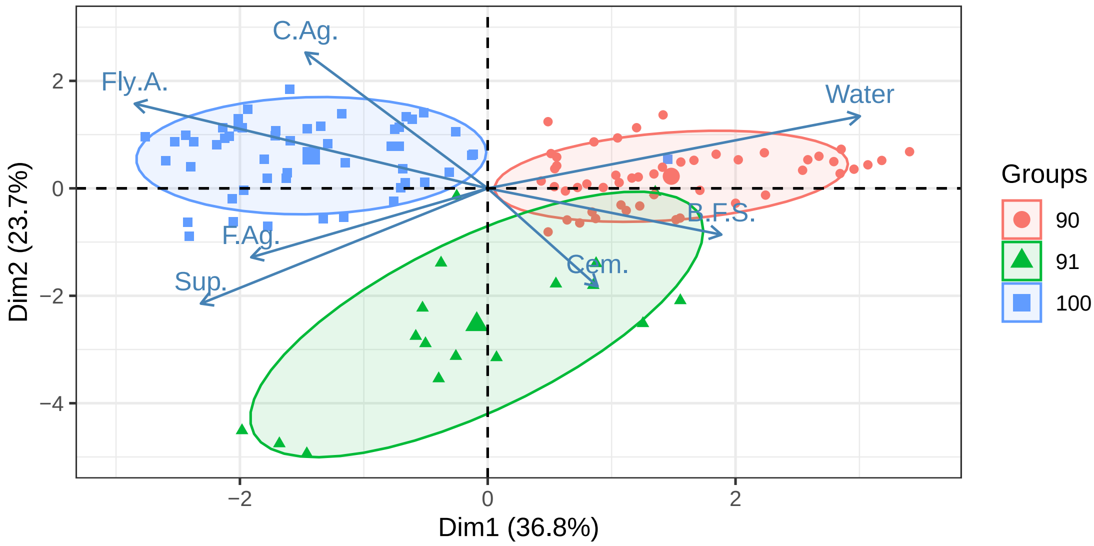

Figure 2. Principal component analysis - 90, 91 and 100 days

After these manipulations, the final number of samples was reduced from 1030 rows to 916 rows, with 416 different ingredient configurations (IDs). A xls file of the data at this point is available for download at the github repository [15].

3.4 Data visualization

In addition to the already presented plots, several other plots and tables were built to perform the exploration and visualization of the samples prepared in the previous step, including:

- Analysis of the descriptive statistics of the continuous and categorical variables;
- Distribution of the variables in relation to the compressive strength;
- Correlation between the variables grouped by age;

- Relationship between the approximate mix and the compressive strength;
- Relationship between the main concrete ingredients and the compressive strength;
- Principal component analysis (PCA) of the ingredients.

The plots for the statistical analysis of the categorical variables are presented, in the Fig. 3.

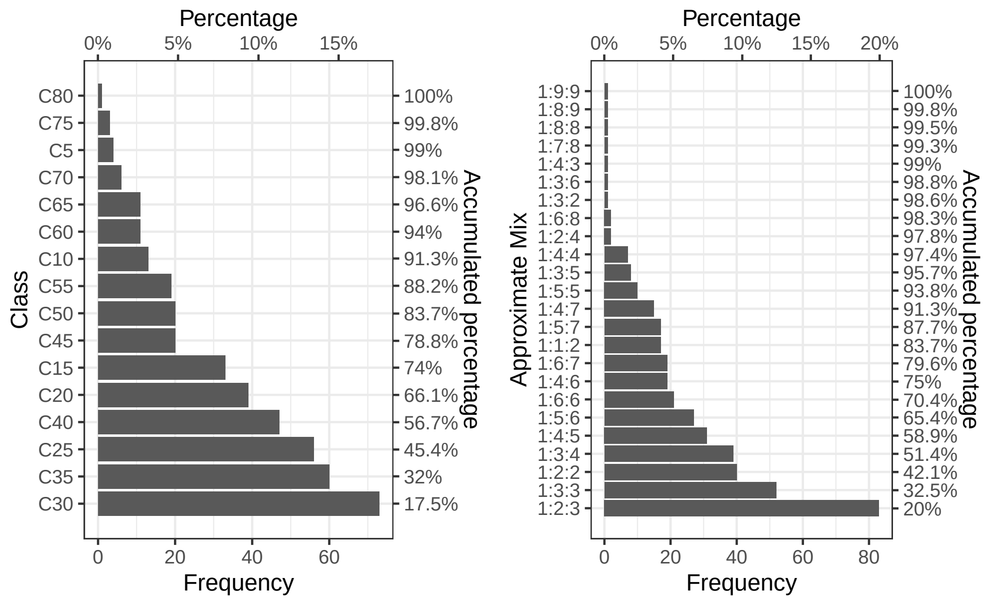

Figure 3. Descriptive statistics of the categorical variables

3.5 Pre-processing and data split

The main package used to build the machine learning models was the Caret Package [12]. It provides all functionalities and utilities to build prediction models for any data set, has a straight and clear documentation that guides the process and provide around 200 different algorithms to build models. In this work, it was done key steps described by Irizarry [17] and Kuhn [18]. Starting by some pre-processing steps described below:

1. Removal of the categorical variables;
2. Separation of the dataset by age, resulting in 6 smaller datasets;
3. Removal of variables with near zero variance (only the fly ash of the 7-day set was removed);
4. Verification of variables with a correlation above 0.999, which did not occur;
5. Each data set was split into test and training sets, 20% and 80% respectively, shown in Table 1 and the distribution in the Fig. 4.

Table 1. Dataset split configurations

Model	Total samples	Train (80%)	Test (20%)
3 days	121	97	24
7 days	114	94	20
14 days	62	50	12
28 days	416	335	81
56 days	83	67	16
100 days	120	96	24

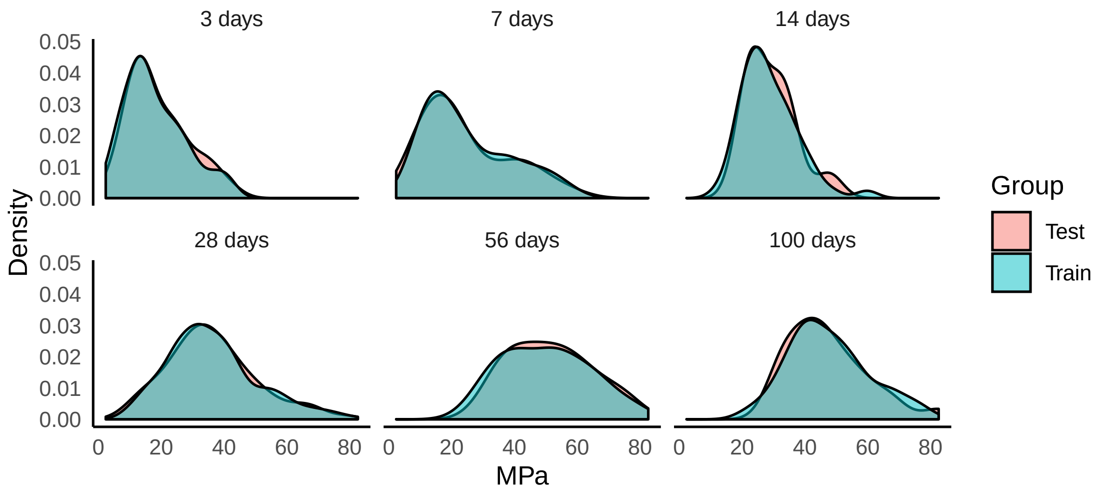

Figure 4. Distribution of the test and train data in relation to compressive strength

3.6 Naive models

Before building the real models, for comparison purposes, naive models were created. They simply predict that the compressive strength of the test set is the average compressive strength of the training set. In other words, naive models are simply the best guess possible to evaluate how close/far the real model is from a guess.

3.7 Machine learning models

We used only one algorithm in this work, chosen by the highest probability to achieve the best possible result, according to Fernandez-Delgado et al. [19], who compared 179 algorithms across 121 different databases, and find out that the most likely to achieve the best possible results is the Parallel Random Forest, called prRF in the Caret Package [12].

Six different models were built, one for each age-set, and the following steps were made for each one:

1. Define the resampling scheme, with method of repeated cross validation;
2. Define a tuning grid for the "mtry" tuning parameter, which is a sequence from 1 to the number of columns of each dataset. All but the 7-day are equal since only the 7-day set have a column removed in the pre-processing;
3. Set seed equal to "1", chosen arbitrarily to guarantee reproducibility. This seed can be manipulated to obtain more satisfactory results, but it was chosen not to.
4. Do pre-processing transformation of the data with "center" and "scale" methods;
5. Run the caret "train" function with the above configurations and model "parRF";

4 Results

The performance evaluation of the models was performed by the Root Mean Square Error (RMSE). The RMSE is the measure used in all the works mentioned in the introduction allowing the comparison of the models in the discussion.

The test RMSE for each model in ascending order of age was 3.31, 4.36, 4.62, 4.72, 5.94 and 5.85 respectively. Table 2 presents the details and results of each model, including the naive one. Fig. 5 compares the actual and predicted values for the final models.

Table 2. Final models results

Model	mtry	CV	Repetitions	Naive RMSE (test)	Final RMSE (train)	Final RMSE (test)
3 days	6	30	10	9.303229	3.905196	3.310370
7 days	2	10	10	13.443646	4.475981	4.361987
14 days	13	30	10	7.593319	5.136687	4.620515
28 days	11	30	10	14.283824	5.847334	4.716698
56 days	8	30	10	12.702112	6.702565	5.939163
100 days	8	10	10	12.614652	6.381940	5.851088

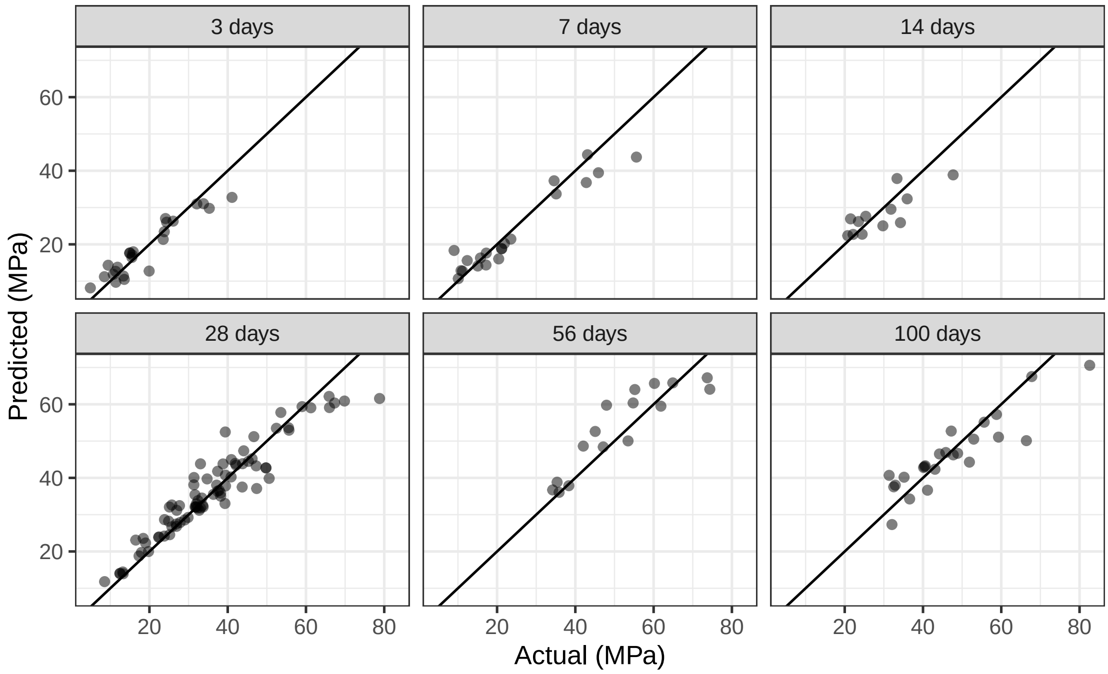

Figure 5. Actual vs predicted values for each model

5 Discussion and conclusion

The built models present satisfactory results and prove that the compressive strength of concrete can be predicted relatively easily. The alternative adopted, creating a model for each set of age proved to be a valid method, instead of using the age as a predictor along with the ingredients like the related studies with the same dataset. The adoption of this stratification achieved different results for each age group. The RMSE calculated in our work and the one obtained in the related works were close. Table 3 shows the comparison between these studies and the 28 days model developed here.

Following the line of reasoning of this work, the same hypothesis can be evaluated using other algorithms besides the one used here (Parallel Random Forest), as they can present better results. Another option is to create an ensemble of various algorithms, just like Pierobon [10], but with the separation of age sets proposed here. In addition, this study can be reproduced with a larger dataset, ideally with a similar number of samples in each age group and a more homogeneous distribution of compressive strength and concrete class, as seen in Fig. 3 that this dataset is very biased to concrete class between C25 and C35.

Table 3. Comparison to other works with same dataset

Author	Year	Algorithm	RMSE	Difference (%)
Pierobon [10]	2018	5 algorithms Ensemble	4.150	-12
This work (28 day)	**2020**	**Parallel Random Forest**	**4.717**	**0**
Hameed and Khalid [6]	2020	Artificial Neural Networks	4.736	0
Raj [8]	2018	Gradient Boosting Regressor	4.957	+5
Modukuru [7]	2020	Random Forest Regressor	5.080	+8
Alshamiri et al. [5]	2020	Regularized Extreme Learning Machine	5.508	+17
Abban [9]	2016	SVM with Radial Basis Function Kernel	6.105	+29

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] Gambhir, M., 1990. *Concrete Technology: Theory and Practice*. Tata McGraw-Hill Education, 5e edition.
[2] Hasan, M. & Kabir, A., 2011. Prediction of compressive strength of concrete from early age test result. In *Conference: 4th Annual Paper Meet and 1st Civil Engineering Congress, At Dhaka, Bangladesh*.
[3] Kabir, A., Hasan, M., & Miah, M., 2012. Predicting 28 days compressive strength of concrete from 7 days test result. In *International Conference on Advances in Design and Construction of Structures (ADCS 2012)*, pp. 18–22.
[4] Yeh, I.-C., 1998. Modeling of strength of high-performance concrete using artificial neural networks. *Cement and Concrete Research*, vol. 28, pp. 1797–1808.
[5] Alshamiri, A., Yuan, T.-F., & Kim, a., 2020. Non-tuned machine learning approach for predicting the compressive strength of high-performance concrete. *Materials*, vol. 13, pp. 1023.
[6] Hameed, M. & Khalid, M., 2020. *Prediction of Compressive Strength of High-Performance Concrete: Hybrid Artificial Intelligence Technique*, pp. 323–335. Spriger.
[7] Modukuru, P., 2020. Concrete compressive strength prediction. https://towardsdatascience.com/concrete-compressive-strength-prediction-using-machine-learning-4a531b3c43f3.
[8] Raj, P., 2018. Predicting compressive strength of concrete. https://www.kaggle.com/pavanraj159/predicting-compressive-strength-of-concrete.
[9] Abban, D., 2016. Concrete compresive strength. https://rpubs.com/brother_abban/220101.
[10] Pierobon, G., 2018. A comprehensive machine learning workflow with multiple modelling using caret and caretensemble in r. https://towardsdatascience.com/a-comprehensive-machine-learning-workflow-with-multiple-modelling-using-caret-and-caretensemble-in-fcbf6d80b5f2.
[11] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al., 2011. Scikit-learn: Machine learning in python. *Journal of machine learning research*, vol. 12, n. Oct, pp. 2825–2830.
[12] Kuhn, M. e. a., 2020. *caret: Classification and Regression Training*. R package version 6.0-86.
[13] R Core Team, 2020. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.
[14] RStudio Team, 2020. *RStudio: Integrated Development Environment for R*. RStudio, Inc., Boston, MA.
[15] Moreira, P. B. A., 2020. Repository. https://github.com/PedroBern/concrete-compressive-strength-prediction.
[16] Yeh, I.-C., 2008. Concrete compressive strength data set archive to download. https://archive.ics.uci.edu/ml/datasets/concrete+compressive+strength.
[17] Irizarry, R., 2019. *Introduction to Data Science: Data Analysis and Prediction Algorithms with R*. Chapman & Hall/CRC Data Science Series. CRC Press.
[18] Kuhn, M., 2008. Building predictive models in r using the caret package. *Journal of Statistical Software, Articles*, vol. 28, n. 5.
[19] Fernandez-Delgado, M., Cernadas, E., Barro, S., & Amorim, D., 2014. Do we need hundreds of classifiers to solve real world classification problems? *Journal of Machine Learning Research*, vol. 15, pp. 3133–3181.