

Computational Analysis of Environmental Risk Conditioners

¹Barbara Carla Coelho Batista, ²João Victor Dutra Balboa, ²Caio Vincenzo Reis Dima, ¹Jordan Henrique de Souza, ²Marcelo Caniato Renhe, ¹Gislaine dos Santos and ²Luciana Conceição Dias Campos

¹School of Civil Engineering, UFJF Federal University of Juiz de Fora, Juiz de Fora (MG), Brazil

²School of Computer Science, UFJF Federal University of Juiz de Fora, Juiz de Fora (MG), Brazil

Article history

Received: 18-06-2020

Revised: 19-09-2020

Accepted: 23-09-2020

Corresponding Author:

Barbara Carla Coelho Batista
School of Civil Engineering,
UFJF Federal University of Juiz
de Fora, Juiz de Fora (MG),
Brazil
Email: nep@engenharia.ufjf.br

Abstract: The intense urbanization process since the 1970s, coupled with the lack of adequate housing and social policies, has led large urban centers to disordered occupations and situations of geotechnical risk. These occupations were not implemented in a technically correct manner from the point of view of civil engineering, considering landscaping, drainage and paving, as well as edification. Areas at risk are regions where it is not recommended to build houses or facilities because they are very exposed to natural disasters, such as landslides and floods. In Brazil, the main institution responsible for monitoring areas at risk is the Civil Defense. There is a large database with history of occurrences of risk areas served by the Municipal Civil Defense, in Juiz de Fora city, Minas Gerais state - Brazil, from 1996 to 2017. Some important information contained in this database are the physical aspects of the soil, such as slope, geolocation, amplitude, curvature and accumulated flow, as well as processed data from the sliding risk susceptibility methodologies. The objective of this work is to apply machine learning techniques to identify, from the mentioned database, the susceptibility to the risk of environmental disasters in regions that have not yet participated in events attended by the municipal civil defense. This database is large and unbalanced, thus it is necessary to apply data analysis methodologies so that the machine learning model can correctly identify the standards with the least human intervention. In this study, areas were classified according to risk susceptibility. After the whole process, it was possible to analyze the performance of the algorithms and select some of them, which obtained the best results, with an accuracy of around 80%.

Keywords: Machine Learning, Environmental Risk, Data Analysis

Introduction

According to the latest survey by the Brazilian Institute of Geography and Statistics (IBGE) in partnership with the National Center for Monitoring and Early Warning of Natural Disasters (CEMADEN), the number of Brazilians living in risk areas, mainly from flooding and landslides, goes over 8 million (Geociencias, 2019). In the city of Juiz de Fora, state of Minas Gerais - Brazil, 25% of the population lives in these conditions (MGTV, 2018), being in the ninth position in the national ranking of cities with the largest number of inhabitants living outside ideal conditions, according to the Statistical Territorial Base of Risk Areas (BATER), obtained by crossing data from the mapping of risk areas, carried out by

CEMADEN and the 2010 Demographic Census of the IBGE (Araujo, 2018). This is due to the accelerated Brazilian urbanization process, as shown in Table 1, with emphasis on the Southeast, the region with the highest rates. According to (Carvalho *et al.*, 2007), risk areas consist of locations susceptible to being affected by phenomena, natural or not, that cause disastrous effect. Therefore, the inhabitants of these areas are subject to physical and material damage.

Allied to the precariousness of many constructions, due to the disorderly process of urbanization, these areas become even more dangerous in terms of disaster risk. For this reason, several efforts are carried out by different public agencies with the aim of preventing these disasters or, at least, citizens' lives.

Table 1: Brazilian urbanization rate (IBGE, 2013)

Region	1940	1950	1960	1970	1980	1991	2000	2007	2010
Brazil	31.24	36.16	44.67	55.92	67.59	75.59	81.23	83.48	84.36
North	27.75	31.49	37.38	45.13	51.65	59.05	69.83	76.43	73.53
Northeast	23.42	26.4	33.89	41.81	50.46	60.65	69.04	71.76	73.13
Southeast	39.42	47.55	57.00	72.68	82.81	88.02	90.52	92.03	92.95
South	27.73	29.5	37.10	44.27	62.41	74.12	80.94	82.9	84.93
Midwest	21.52	24.38	34.22	48.04	67.79	81.28	86.73	86.81	88.8

The work in question aims to assist in this prevention, using machine learning techniques, in order to carry out the classification of areas according to their risk susceptibility. The main objective is to identify regions with a high likelihood of disasters occurring, even though no events have previously been recorded in these regions. The classification is performed using data from regions with known occurrences, thus relating the characteristics of the classified place with others that have properties in common. As a secondary objective, the work looks for patterns in the data that explain and determine what leads a region to be considered a risk area.

Bibliographic Review

According to (Samuel, 2000), machine learning is defined as the ability of computers to learn without being explicitly programmed, that is, it is based on the idea that systems can learn from data, identifying patterns with minimal human intervention. A widely used machine learning technique is Artificial Neural Networks (ANN). According to (Haykin, 2007), an ANN consists of the interconnection of processing units (artificial neurons) that store knowledge through a learning process, which consists of adjusting the weights of the network interconnections (synaptic weights), making them available for use. Figure 1 shows an ANN model. These are non-linear structures that adapt according to training and their functioning is inspired by the human brain.

Algorithm Performance Metrics

There are several Classification Algorithms in the literature that can be used to solve problems of this nature. Section “Classification” presents the algorithms used in this study. However, there is no classification technique in the literature that is applicable to all problems that occur in practice. That is, an algorithm can effectively solve a given problem and not have the same capacity to solve another problem, even if it is apparently similar to the previous one. Thus, it is necessary to make an efficiency comparison between the different systems used. Generally, a set of algorithms is evaluated, in order to select the one that obtained the best performance in solving a problem, using different evaluation metrics.

In the classification algorithms, it is very common to use the confusion matrix as a performance analysis. This matrix shows the convergence and divergence between the real values and the predictions made (responses from the classifying algorithms) (Ting, 2017). It is a square matrix, in which the order is the number of classes of the problem. In this case, there are two classes for risk: One positive and one negative. Therefore, as the classification is binary, there is a special case of matrix, as shown in Fig. 2, in which the first column is related to the negative classifications generated by the algorithm and the second is related to positive ones. In relation to the lines, the first represents the real negative values while the second is associated with the real positive values. Thus, the intersections represent the hits and errors (e.g., the values in the negative column and negative line mean true negative, that is, the model correctly classified those entries as having no risk) and each position in the matrix has a special name, being: True Negative (TN), True Positive, (TP) False Negative (FN) and False Positive (FP).

Using the confusion matrix, the following metrics can be calculated:

- **Accuracy:** Analyzes the correctness (*TN* and *TP*) of the classifier. Its calculation is shown in Equation 1, where total represents the total data:

$$\frac{TP + TN}{total} \tag{1}$$

Where:
TP = True Positive
TN = True Negative

- **Precision:** Shows the proportion of positive cases classified in this way, taking into account all positive classifications made by the algorithm, as shown in Equation 2:

$$\frac{TP}{TP + FP} \tag{2}$$

Where:
TP = True Positive
FP = False Positive

- **Specificity:** Indicates the proportion of correctly classified cases (TN), that is, the proportion of true negatives related to the relationship with the total number of caused particles attracted by Equation 3:

$$\frac{TN}{TN + FP} \quad (3)$$

Where:

TN = True Negative

FP = False Positive

- **Sensitivity (Recall):** represents the ability to correctly classify positive cases, according to the existing positives, as shown in Equation 4:

$$\frac{TP}{TP + FN} \quad (4)$$

Where:

TP = True Positive

FN = False Negative

- **F1-score:** Expresses the balance between precision and sensitivity, through the harmonic mean between them. It ranges from 0 to 1 and, the higher, the better these two metrics are. Equation 5 is used to calculate this metric:

$$F1score = \frac{Precision \cdot Recall}{Precision + Recall} \quad (5)$$

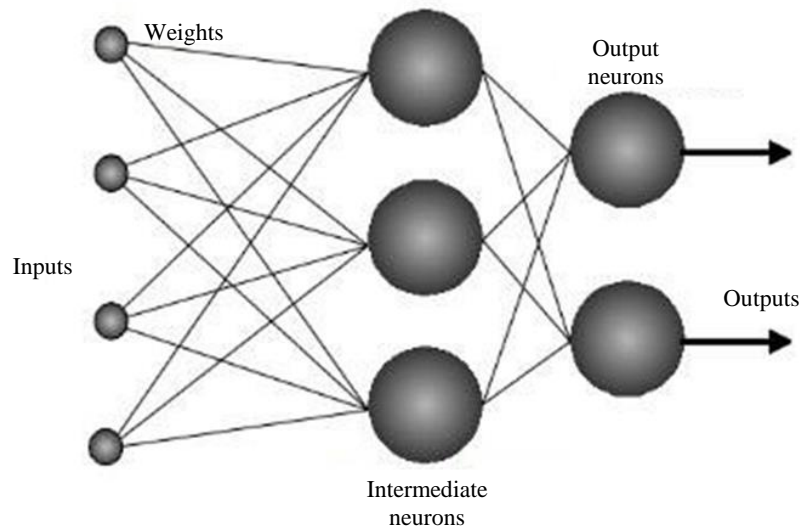


Fig. 1: Model of an Artificial Neural Network (Tafner, 1998)

CONFUSION MATRIX		
	Test Negative	Test Positive
Condition Negative	TRUE NEGATIVE	FALSE POSITIVE
Condition Positive	FALSE NEGATIVE	TRUE POSITIVE

Fig. 2: Confusion matrix (Mosconi, 2017)

Related Works

Ni *et al.* (1996), in his work, uses Neural Networks in combination with Fuzzy Logic to assess slope stability. The used input variables are divided into 4 categories: Topography, geology, environment and meteorology, these variables have several factors. Altogether, there are 13 factors: Slope, horizontal cut, vertical cut, location, height, geological origin, soil texture, weathering depth, vegetation, land use, maximum daily precipitation and maximum hourly precipitation. The ANN output is the potential for slope failure. This model produced results comparable to those of an analytical method, normally used to assess slope stability.

In the work of (Ferentinou and Sakellariou, 2007), the backpropagation algorithm, the theory of Bayesian neural networks and Kohonen's self-organizing maps are used to predict slope stability. The entry is summarized in altitude, average annual precipitation, slope, lithology, depth of the surface fault and the movement classification. As outputs, there are safety and stability factors. The results have shown promise for further studies.

Gordan *et al.* (2016) propose a combination of particle swarm optimization and neural network to predict slope stability during earthquakes. The input variables are: Slope height, slope, cohesion, friction angle and ground acceleration. The output variable is the safety factor. The results show that particle swarm optimization surpasses simple neural networks in relation to precision.

What differs the work in question from the others mentioned is the fact that it considers several types of risk (the main ones being landslides and flooding), despite having a greater focus on risk on slopes, which is the only one considered in other works. Therefore, it is possible to make the risk classification for all locations. This makes the inputs different, as well as the output, which in this case, is the result of a binary classification, with positive or negative responses to the risk.

Disasters such as landslides and floods are related not only to environmental phenomena and climate change but also to urbanization without planning. For this

reason, it is necessary to employment of public policies that act in urban planning, avoiding the construction of houses without technical support and in risk areas, for example (UFDJDF, 2018).

Methodology

The work consists of applying machine learning techniques to classify some mapped regions, without any occurrence of disasters, as being areas of risk or not. Figure 3 shows the division of the work steps. The first step consists of collecting and pre-processing the data. Once the database is obtained, it is used to train the classification algorithms. Finally, an assessment is made of the obtained results using the performance metrics presented in section "Algorithm Performance Metrics".

Pre-Processing and Data Collection

There is a large database with a history of occurrences of risk areas attended by the Municipal Civil Defense, in the city of Juiz de Fora, from 1996 to 2017. This database has 80,545 records with the following characteristics: Date, place and type of occurrence attended.

The first stage of the work consisted of pre-processing this database, aiming to eliminate missing, inconsistent or incorrect data. After this cleaning, the base now has 43,089 records.

Through the addresses, present at the location of each record, the Geocoding algorithm was used (Vilimpoc, 2019) to obtain the latitude and longitude coordinates that represent the exact point of occurrence. Figure 4 shows the distribution of occurrences (dots in red) across the city of Juiz de Fora.

In a project by NASFE - Center for Social Assistance of the Faculty of Engineering - Federal University of Juiz de Fora, the Municipality's Allotment Plan was used as a basis for dividing the city into lots in ArcGIS software (version 10.5) - geographic information, for processing and extraction of soil aspects (Freitas, 2011) - obtaining a database with 103,651 mapped lots.

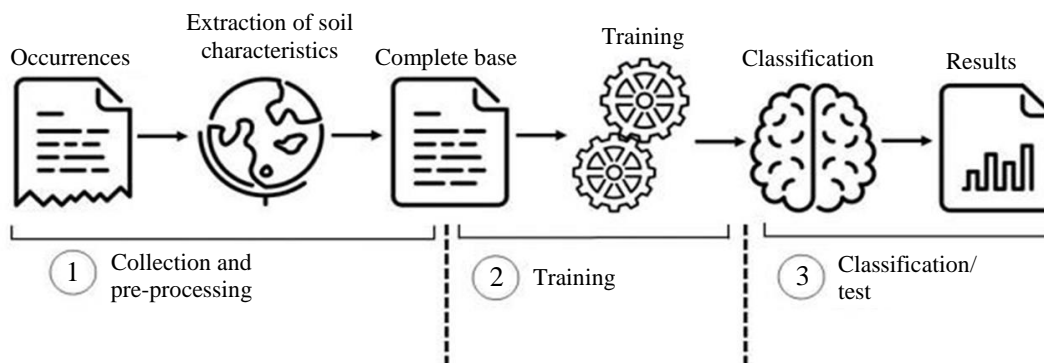


Fig. 3: Work steps



Fig. 4: Civil defense occurrences in Juiz de fora

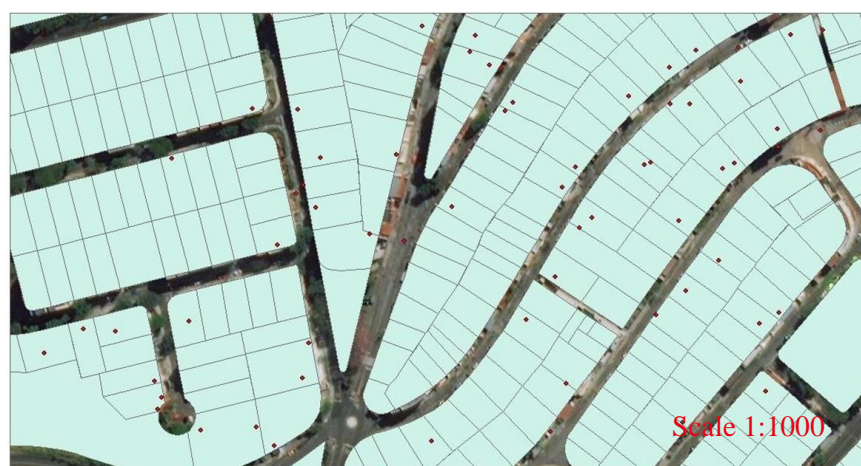


Fig. 5: Delimitation of lots and distribution of occurrences

Table 2: Characteristics of the soil that make up the base

Physical aspects	Methodologies
Declivity	Shalstab
Accumulated flow	IPT- instituto de pesquisas tecnológicas
Altimetry	SAGA
Amplitude	Smorph
Curvature	Smorph modificado
Slope orientation	

In the next step, the coordinates of the occurrences and lots were crossed, generating a binary control variable, which indicates the existence or not of occurrences in each lot. Figure 5 shows the division of the lots together with the occurrences (note that in some there is an occurrence and in others not). A margin of 3 meters was defined to consider if the event happened inside the lot in question, since there were some occurrences outside of them (e.g., occurrences located in front of the lots, on streets etc.).

After this step, it was possible to extract the characteristics of each of the lots (Table 2), which started to compose the database together with the variable control torque. Only the lots that have occurrences were considered as risk areas in the database, which is used in the training of the classification algorithms, as can be seen in Fig. 3.

The pre-processing of data is fundamental to the process of classifying them. As the database has 103,651 mapped lots and we consider 43,089 records of occurrences, the first thing to do in the database was to balance the number of data that present a risk with those that do not. After that, before using it in the training of the classification algorithms, a normalization of this was done - a method that reduces the values of the characteristics of the base to a common scale, without distorting the differences between them.

The used normalization method was the min-max, in which the maximum value of each characteristic is transformed into 1, the minimum in 0 and the others in

decimals between 0 and 1. Equation 6 shows how this normalization is done:

$$x = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \quad (6)$$

Where:

- x^j = Normalized value
- x = Current value
- x_{\max} = Maximum value of the characteristic in question
- x_{\min} = Minimum value

Classification

The technologies used in the development of data classification models are summarized in the Python language (version 3.7.1), with their development environments (Spyder 3.3.4 and Jupyter 5.7.4, made available by the data platform Anaconda 1.9.6) and the Scikit-Learn library (Pedregosa *et al.*, 2011), which has several algorithms for data analysis and prediction.

This subsection represents steps 2 and 3 of the work (Fig. 3). In the training process, a Classification model is applied to learn, in the best possible way, a pattern that can identify the regions that are at risk and those that are not. The lots in which some type of disaster occurred are considered to be areas of risk. Therefore, the database used in the training contains the expected output (binary variable that represents the presence or absence of a disaster occurring in the

region). Thus, the adjustment of the classification model is done taking into account the generated error, calculated by subtracting the model's output from the actual response present in the database. This type of training is called Supervised Training and the generated error represents a model performance metric.

In the training phase, a large part of the database is used (around 2/3) and the remaining part is left for the test phase of the model. The test phase occurs after the training phase, in which the classification model is used to classify the part of the data that was not used in the training, that is, data that is unknown to the model. Through the testing phase, the model's ability to find patterns and similarities between the data used in the training and the unknown data is verified. The performance analysis of the classification models uses the metrics defined in section "Algorithm Performance Metrics". Finally, the models that obtain the best metrics with the results are used to classify areas for which there is no information on risk susceptibility.

In steps 2 and 3 of the work (Fig. 3), different machine learning algorithms were used for data classification, all made available by the Scikit-Learn library. Table 3 presents the different algorithms used and a brief description of each one.

Note 1: Many of the algorithms described in Table 3 have versions for regression and other machine learning problems. Therefore, in the representation, its settings for classification problems were considered.

Table 3: Algorithms used at work (Pedregosa *et al.*, 2011)

Algorithm	Description
K Neighbors	Non-parametric method, in which the input consists of the k closest observations and the output is a grouping of classes. An object is classified by a plurality of votes from its neighbors, being attributed to the most common class among its closer k neighbors
Gradient Boosting	Technique that consists of building a forecasting model using a set of "weak" models and converting them into "strong" models. Decision trees are used and the weights of the observations are adjusted, according to their level of difficulty, after the evaluation of each set of trees. After this, new sets are generated with the previous trees and, therefore, for each new set, its forecast is the weighted sum of the predictions generated by the previous ones
MLP (Multi-layer Perceptron)	It is a Supervised Learning algorithm that, given a set of characteristics (features) $X = x_1, x_2, \dots, x_m$ and a target(target) y , it learns a non-linear approach function $f(): R^m \rightarrow R^o$, training the data set with the Back-propagation algorithm, where m is the number of input dimensions and o is the number of output dimensions
Naive Bayes	The Naive Bayes method is a set of Supervised Learning algorithms based on the Bayes Theorem with an ingenuous presumption that each feature has its value independent of the others
Decision Tree	It is a nonparametric Supervised Learning method, the purpose of which is to create a model that predicts the value of a target, learning simple decision rules inferred from the data resources
Random Forest	Estimator method that, in summary, builds a multiplicity of decision trees at the time of training and uses their averages to improve the accuracy of the classification
SGD	It is a discriminative learning algorithm for linear classification under convex loss functions, such as Support Vector Machines and linear regression. It implements a simple stochastic gradient descent learning routine that supports different loss functions and penalties for classification
One Vs Rest	The method consists of fitting one classifier per class. For each classifier, the class is adjusted in relation to all others. Since each class is represented by one and only one classifier, it is possible to know the class by inspecting its corresponding classifier

Results and Discussion

Among the algorithms present in Table 3, the ones that stood out the most were: K Neighbors (with a K value of 2), Gradient Boosting and MLP, which obtained the best values in metrics, such as accuracy, represented in Fig. 6. Note in the figure that all algorithms achieved accuracy greater than 50% and that the aforementioned algorithms achieved an accuracy around 80%.

Figure 7 shows the confusion matrix, which is another way of analyzing the results of the 3 best algorithms in relation to accuracy. This shows the hits and errors generated, the false negative being the main error to be avoided, in which the classification made does not consider that the area is at risk, but its real condition is at risk. Note that the K Neighbors algorithm is the one that most hits the positive cases and is the one that misses the false negative the least. The MLP algorithm hits the negative cases the most and does not miss any false positive, however it is the one that misses the most false negative cases and that least hits the positive ones. Therefore, for this work problem, MLP performs worse than K Neighbors.

Table 4 presents the performance metrics of all the algorithms in Table 3.

Comparing the 3 algorithms that had the best accuracy - K Neighbors, Gradient Boosting and MLP, it is possible to observe that the MLP algorithm reached 100% in the metrics of precision and specificity, which is in line with its confusion matrix, since MLP had no case of FP (False Positive). However, when analyzing the sensitivity meter, which uses FN (False Negative), MLP presents the lowest performance in the table. The K Neighbors algorithm, in turn, presents the best accuracy and has a high sensitivity, being therefore considered the chosen one as the best classifier for this work database.

Figure 8 the comparison between the classification made by the K Neighbors algorithm and the real condition of each lot. Red lots represent positive for risk and green ones represent negative. Note that the forecast is able to correctly classify most of the lots.

Figure 9 the classification made by the K Neighbors algorithm, where the lots are colored according to the successes and errors represented in the confusion matrix of Fig. 7a. It is possible to notice in the figure that the correctness overcomes the classification errors, reinforcing that the K Neighbors performed well with this database.

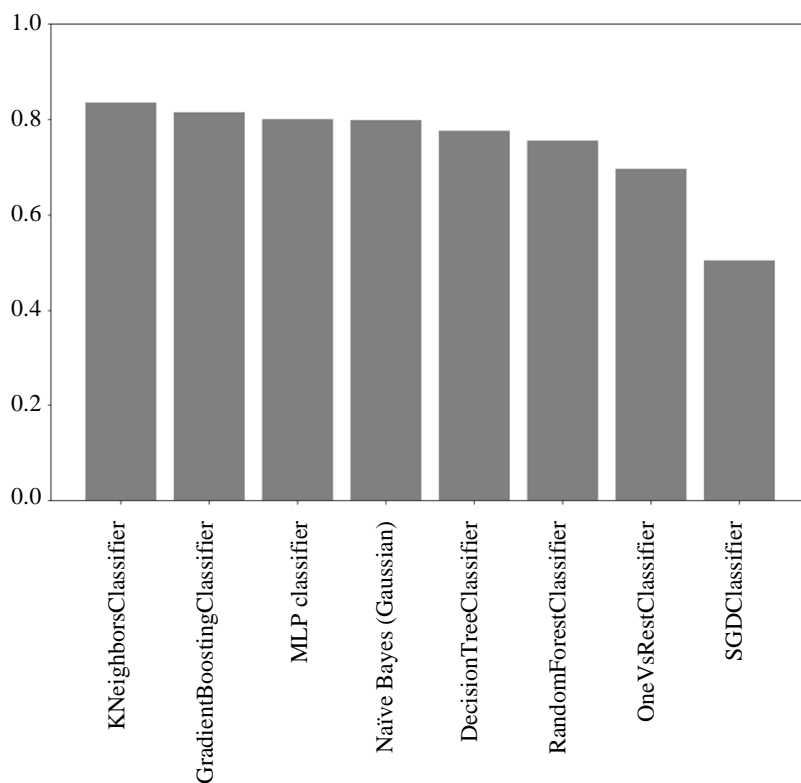


Fig. 6: Comparison of accuracy between algorithms

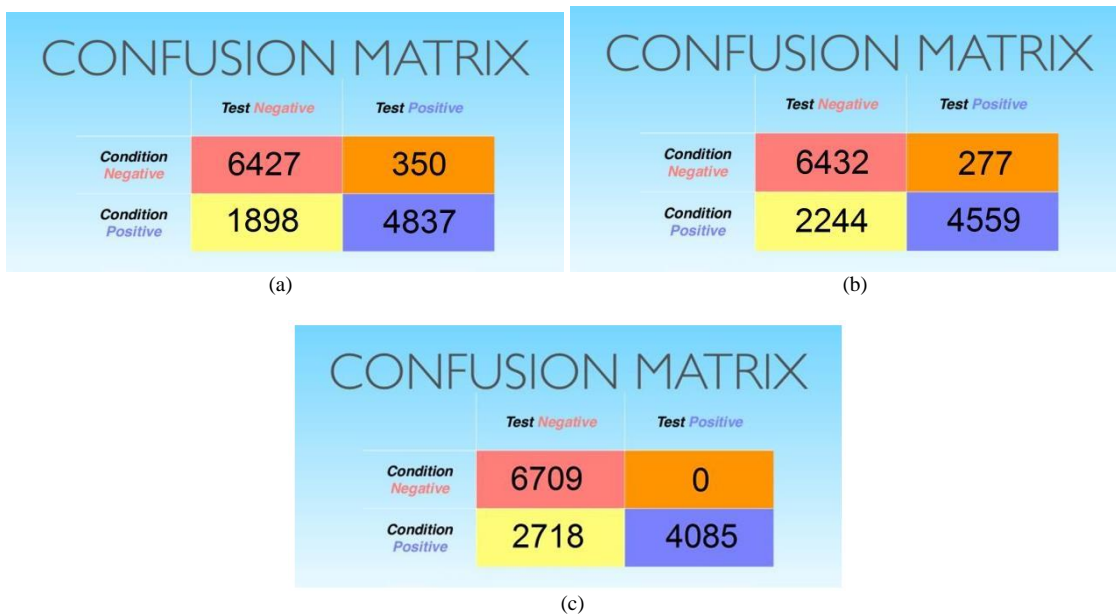


Fig. 7: Confusion matrix of classifiers; (a) K Neighbors; (b) Gradient Boosting; (c) MLP

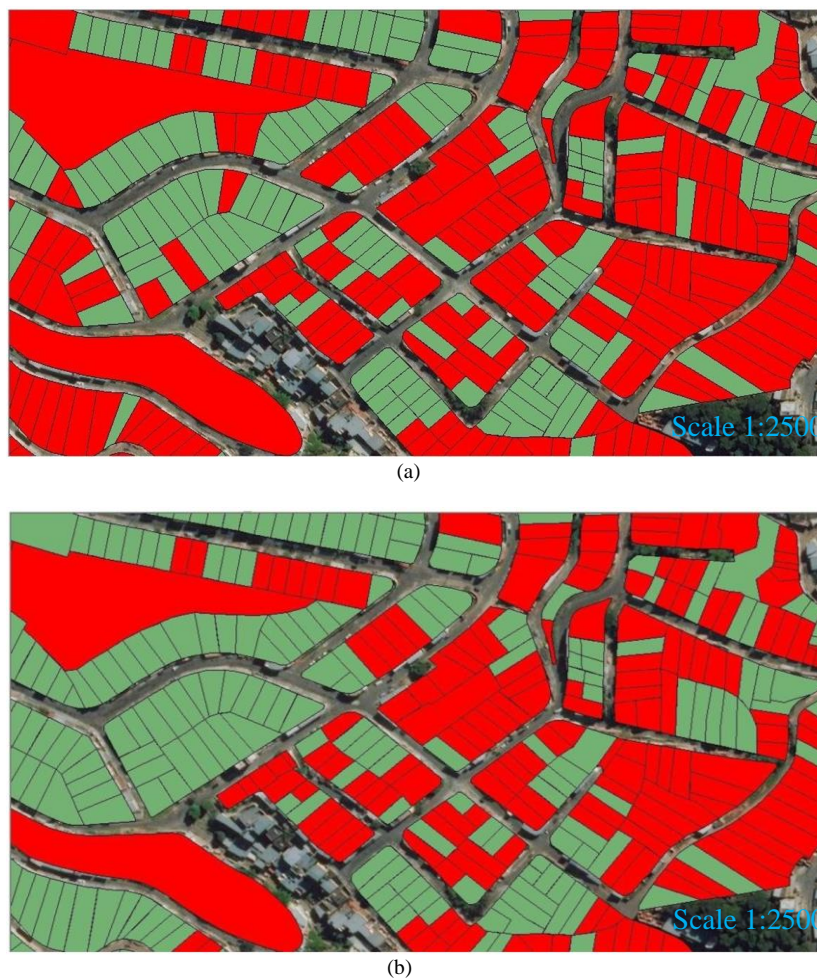


Fig. 8: Classification of part of juiz de for a; (a) Real; (b) Prevision



Fig. 9: Classification of a fraction of lots in the city of Juiz de Fora, with labels from the Confusion Matrix of the K Neighbors

Table 4: Performance metrics

Algorithm	Accuracy	Precision	Specificity	Sensitivity	F1-score
K Neighbors	0.834	0.880	0.898	0.750	0.810
Gradient Boosting	0.813	0.943	0.959	0.670	0.783
MLP	0.799	1.000	1.000	0.600	0.750
Naive Bayes	0.798	0.955	0.970	0.629	0.759
Decision Tree	0.775	0.774	0.769	0.780	0.777
Random Forest	0.756	0.787	0.805	0.707	0.745
SGD	0.503	0.503	0.000	1.000	0.670
One Vs Rest	0.696	0.694	0.684	0.707	0.701

Conclusion

In this study, areas were classified according to risk susceptibility. After the whole process, it was possible to analyze the performance of the algorithms and select some of them, which obtained the best results, with accuracy of around 80%, to, from then on, generate the classification of new areas, in a future work.

It was concluded that it is feasible to classify the areas as they were done, although further studies are still needed to reduce the error in the generated classification, especially the false negative, which is the most dangerous in the case in question.

In addition, it is intended to carry out the classification of the type of existing risk (e.g., landslide, flood, etc.), as well as to calculate the probability of occurrence of each type, since the classification is so far purely binary (whether or not it has the risk).

Another future work would be to integrate the created model with an application for mapping risk areas developed within the scope of the Federal University of Juiz de Fora (Álea) (De Souza, 2018), thus being able to

suggest the degree of risk when entering information from a location. The application in question is aimed at professionals from the Fire Department and Civil Defense and has several forms for filling out information collected in the field, also allowing the manual delimitation of the analyzed area by means of marking polygons. Having this delimitation, it is possible to extract the soil aspects used in this study and suggest a degree of risk for the user who is registering the area, or even classify it automatically.

The present work demonstrates the need to use computational tools, as well as the use of artificial intelligence to analyze the complex risk scenario in urban areas. The 80% assertiveness in the prediction of environmental risk demonstrates the importance of these tools for disaster risk reduction. The concepts and tools presented in this study can be replicated in other areas. As a recommendation for future work, it is recommended to include the variable amount of rainfall in the last hours, so that the assertiveness of the computational model can be increased.

Acknowledgment

The authors generously thank the Federal University of Juiz de Fora (UFJF), in particular the Pro-rectories, both Graduation (PROGRAD) and Extension (PROEX), for the scholarships granted, which allowed the studies for the preparation of this work. To the Departments of Computer Science (DCC) and Transport and Geotechnics (TRN) and also to the 3rd Operational Fire Command (3rd COB) of the Minas Gerais Military Fire Brigade (CBMMG) and the Municipal Civil Protection and Defense Coordination (COMPDEC) of Juiz de Fora, both for the partnership in the studies. We also would like to thank the UFJF International Relations Office for the linguistic revision of the manuscript.

Funding Information

This research was developed by the School of Civil Engineering and School of Computer Science at the Federal University of Juiz de Fora.

Author's Contributions

Barbara Carla Coelho Batista and Joao Victor Dutra Balboa: Wrote the article as a leading author. Designed the research plan, participated in all programming and calculation, coordinated the data analysis and contributed to the writing of the paper.

Caio Vincenzo Reis Dima: Organized the study, participated in all experiments, programming and calculation and contributed to the writing of the paper.

Jordan Henrique de Souza and Marcelo Caniato Renhe: Participated in all experiments and in the data analysis also contributing to the writing of the paper and the data analysis.

Gislaine dos Santos and Luciana Conceição Dias Campos: Participated in all experiments and in the data analysis also contributing to the writing of the paper and the data analysis. Critically reviewed the manuscript.

Ethics

This article is original and contains unpublished material. The corresponding author confirms that the other authors have read and approved the manuscript and that no ethical issues are involved.

References

Araujo, M. (2018). Jf tem quase 130 mil pessoas vivendo em áreas de risco. <https://tribunademinas.com.br/noticias/cidade/13-09-2018/jf-tem-quase-130-mil-pessoas-vivendo-em-areas-de-risco.html>

- Carvalho, C. S., Macedo, E. S. D., & Ogura, A. T. (2007). Mapeamento de riscos em encostas e margem de rios. Brasília: Ministério das Cidades.
- De Souza, J. H. (2018). Mapeamento De áreas De Risco Com O Aplicativo Álea. Clube de Autores (managed).
- Ferentinou, M. D., & Sakellariou, M. G. (2007). Computational intelligence tools for the prediction of slope performance. *Computers and Geotechnics*, 34(5), 362-384.
- Freitas, E. (2011). ArcGIS10. <https://mundogeo.com/2010/08/05/arcgis-10/>
- Geociencias, (2019). Em 2010, brasil tinha 8,3 milhões de pessoas morando em áreas com risco de desastres naturais. <https://agenciadenoticias.ibge.gov.br/agencia-sala-de-imprensa/2013-agencia-de-noticias/releases/21565-em-2010-brasil-tinha-8-3-milhoes-de-pessoas-morando-em-areas-com-risco-de-desastres-naturais>
- Gordan, B., Armaghani, D. J., Hajihassani, M., & Monjezi, M. (2016). Prediction of seismic slope stability through combination of particle swarm optimization and neural network. *Engineering with Computers*, 32(1), 85-97.
- Haykin, S. (2007). *Redes neurais: Princípios e prática*. Bookman Editora.
- IBGE. (2013). Atlas do censo demográfico 2010.
- MGTV, (2018). Pelo menos 25% dos moradores de juiz de fora vivem em área de risco, aponta levantamento. <https://g1.globo.com/mg/zona-da-mata/noticia/pelo-menos-25-dos-moradores-de-juiz-de-fora-vivem-em-area-de-risco-aponta-levantamento.ghtml>
- Mosconi, F. (2017). Machine learning: Techniques, best practices and practical application. <https://pt.slideshare.net/FrancescoMosconi/machine-learning-techniques-best-practices-and-practical-application>
- Ni, S. H., Lu, P. C., & Juang, C. H. (1996). A fuzzy neural network approach to evaluation of slope failure potential. *Computer-Aided Civil and Infrastructure Engineering*, 11(1), 59-66.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O.,... & Vanderplas, J. (2011). Scikit-learn: Machine learning in Python. *The Journal of machine Learning research*, 12, 2825-2830.
- Samuel, A. L. (2000). Some studies in machine learning using the game of checkers. *IBM Journal of research and development*, 44(1.2), 206-226.

- Tafner, M. A. (1998). Redes neurais artificiais: Aprendizado e plasticidade. <http://www.cerebromente.org.br/n05/tecnologia/plasticidade2.html>
- Ting, K. M. (2017). Confusion Matrix. Encyclopedia of Machine Learning and Data Mining, 260.
- UFDJDF. (2018). Mapeamento De áreas De Risco Com O Aplicativo Álea. Juiz de Fora: Universidade Federal de Juiz de For a.
- Vilimpoc, M. (2019). Google Sheets Geocoding Macro. GitHub, <https://github.com/nuket/google-sheets-geocoding-macro/blob/master/README.md>