



# Comparison of machine learning tools for damage classification: the case of L'Aquila 2009 earthquake

F. Di Michele<sup>1</sup> · E. Stagnini<sup>1</sup> · D. Pera<sup>2</sup> · B. Rubino<sup>2</sup> · R. Aloisio<sup>1,3</sup> · A. Askan<sup>4</sup> · P. Marcati<sup>1</sup>

Received: 2 June 2022 / Accepted: 24 December 2022  
© The Author(s) 2023

## Abstract

On April 6, 2009, a strong earthquake (6.1 Mw) struck the city of L'Aquila, which was severely damaged as well as many neighboring towns. After this event, a digital model of the region affected by the earthquake was built and a large amount of data was collected and made available. This allowed us to obtain a very detailed dataset that accurately describes a typical historic city in central Italy. Building on this work, we propose a study that employs machine learning (ML) tools to predict damage to buildings after the 2009 earthquake. The used dataset, in its original form, contains 21 features, in addition to the target variable which is the level of damage. We are able to differentiate between light, moderate and heavy damage with an accuracy of 59%, by using the Random Forest (RF) algorithm. The level of accuracy remains almost stable using only the 12 features selected by the Boruta algorithm. In both cases, the RF tool showed an excellent ability to distinguish between moderate-heavy and light damage: around the 3% of the buildings classified as seriously damaged were labeled by the algorithm as minor damage.

**Keywords** Seismic damage prediction · Machine learning · L'Aquila 2009 earthquake · Ground motion

## 1 Introduction

The city of L' Aquila, Italy, and several surrounding towns were completely leveled by an earthquake on the morning of April 6, 2009, at 3:32 a.m. (6.1 Mw). The quake happened on a typical fault that runs northwest to southeast. The hypocenter was pinpointed

---

✉ F. Di Michele  
federica.dimichele@gssi.it

<sup>1</sup> Gran Sasso Science Institute (GSSI), via M. Iacobucci 2, L'Aquila, Italy

<sup>2</sup> Department of Information Engineering, Computer Science and Mathematics, University of L'Aquila, via Vetoio, loc. Coppito, I, L'Aquila, Italy

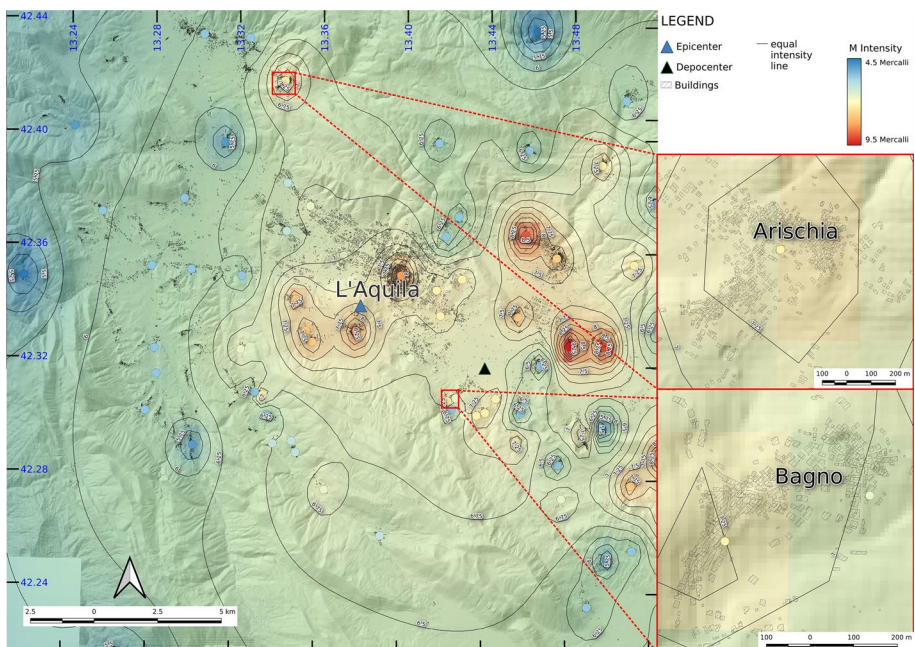
<sup>3</sup> INFN-Laboratori Nazionali del Gran Sasso, Via G. Acitelli 22, Assergi, AQ, Italy

<sup>4</sup> Departments of Civil Engineering and Earthquake Studies, Middle East Technical University, 06800 Çankaya, Ankara, Turkey

at a depth of 8 kilometers, not far from the town of Roio (42.34 degrees N latitude, 13.37 degrees E longitude; <http://terremoti.ingv.it/>). L'Aquila is a bustling city that is home to a wide variety of buildings, both modern and ancient. Most of the homes were built at a time when construction codes were different.

As a result of the earthquake's directivity effects, stations aligned with the rupture direction registered ground vibrations of high amplitude and short duration, whereas stations aligned with the opposite direction registered ground motions of low amplitude and long duration. More than 300 people were killed and over 1600 were injured, despite the fact that this earthquake was only moderate in size (Ugurhan et al. 2012). This was because directivity effects combined with regional building characteristics caused catastrophic devastation in the meizoseismal area in the city.

Felt (macroseismic) intensity is one qualitative measure that can be used to quantify the impact of ground shaking caused by earthquakes. These numbers are derived from correlations between felt intensity and measurable ground motion parameters or from human reactions to ground shaking and damage to nearby structures. Damage after an earthquake can be visualized through the use of felt intensity metrics. Figure 1 shows that the communities of Castelnuovo and Onna registered the highest measured intensity values after the L'Aquila event (IMCS = IX – X) (Rovida et al. 2020, 2022). The central business district of L'Aquila and other areas in the epicenter's immediate vicinity show signs of severe damage with IMCS values over VII. Many commercial and residential structures kept crumbling or being severely damaged. Research on the ability to anticipate and categorize earthquake-induced structural damage is therefore crucial. The results of such an attempt would be useful not only in predicting how badly future earthquakes in the area would destroy buildings, but also in understanding the distribution of damages caused by the L'Aquila quake.



**Fig. 1** Microseismic intensity distribution obtained from Rovida et al. (2020, 2022)

After the earthquake, in parallel to a long and complex strengthening and reconstruction process, a massive data acquisition and sharing activity began, followed by the launch of the following two platforms:

- *Open Data L'Aquila* (GSSI 2019a).
- *Open Data Ricostruzione* (GSSI 2019b).

The first repository is mostly concerned with the reconstruction process of L'Aquila city and of surrounding villages. It contains data and maps which are continuously updated. The second one focuses on the municipality of L'Aquila and contains high-resolution maps, solar radiance, and water accumulation maps, an earthquake simulator as well as much other information regarding the geophysical and urban aspects of the city. Drones were used for the aero-photogrammetric survey, which allowed the production of extremely detailed geo-referenced maps and digital models as shown in Fig. 2.

The standard approach to *estimate* the possible damage to a building after an earthquake involves the use of fragility curves. These curves assign, based on statistical tools, the probability that a specific building type exceeds a certain damage threshold under various levels of ground motions. Using these curves, it is possible to estimate the probabilities of the occurrence of predefined damage states given the recorded ground motions of a seismic event. The literature in this context is very broad, see for example Kirçil and Polat (2006), Shinozuka et al. (2000), Lallemand et al. (2015), Baker (2015) and De Luca et al. (2015), Del Gaudio et al. (2019), Rosti et al. (2018), mostly focused on the 2009 L'Aquila earthquake.

In this article, machine learning tools able to estimate the damage to buildings after an earthquake are described. This approach, potentially complementary to the aforementioned fragility curves, was initially first proposed in Mangalathu and Jeon (2018), where the authors applied four different supervised machine learning techniques to predict the level of earthquake-induced damage to structures. They used a dataset consisting of 3677 buildings (reduced to 2276 after the data-cleaning procedure) located in the cities of Napa, American Canyon, and Vallejo, that were struck by a 6.0 Mw magnitude earthquake on August 24, 2014. In that study, each building has been assigned

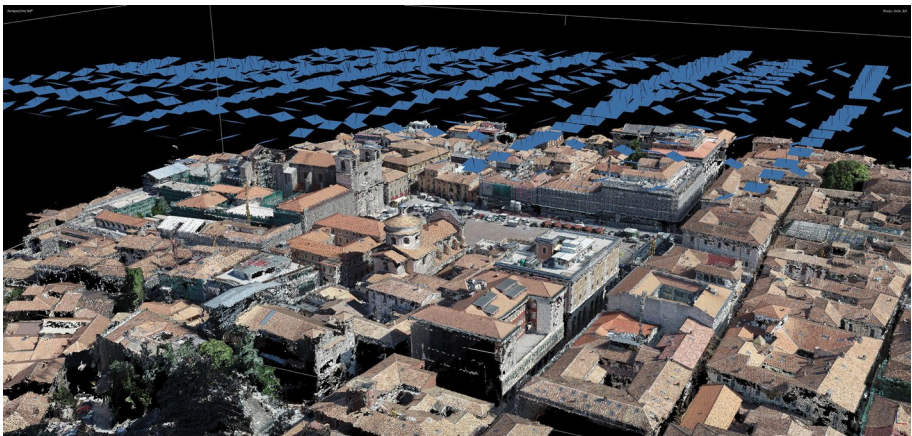


Fig. 2 Dense point cloud from aero-photogrammetric drone survey and processing

the spectral acceleration at 0.3 s at that location (INGV 2020), distance from the projection of the fault plane, shear wave velocity at 30 m, the number of floor levels, the age, building irregularity, footprint of the building, and building economic value. The target value was the level of damage: red, yellow, and green for heavy, moderate, and light damage levels, respectively. The accuracy reached 66% for the Random Forest algorithm, but the dataset in that study contains only 81 red-tagged buildings.

A similar approach is also used in Harirchian et al. (2021) and Roeslin et al. (2020). In Harirchian et al. (2021), the authors applied four different machine-learning techniques to evaluate seismic damage. Four different earthquakes have been taken into account: Ecuador 2016 ( $M_w = 7.8$ ) with 172 buildings, Haiti 2017 ( $M_w = 7.0$ ) with 145 buildings, Nepal 2015 ( $M_w = 7.3$ ) with 135 buildings, and South Korea 2017 ( $M_w = 5.5$ ) with 74 buildings. According to their numerical results, supervised learning provides better prediction rates.

Roeslin et al. (2020) have studied a dataset containing 340 buildings in the Mexico City urban area in relation to the 2017 Mexico City earthquake ( $M_w = 7.1$ ). For each building, 12 features have been assigned including geographical coordinates, construction techniques (structural system such as reinforced concrete frames, masonry, etc.), Peak Ground Acceleration (PGA), and the number of stories. The target variable is the damage rates assigned as *negligible to light damage* and *moderate damage to collapse*. Among the employed algorithms including Logistic Regression, Support Vector Machine, Decision Tree, and Random Forest, the last one achieves the best level of damage prediction accuracy which is 0.65 (65%).

In Yerlikaya-Özkurt and Askan (2020), the authors consider a dataset containing 228 reinforced concrete frame buildings, damaged during three different earthquakes that occurred in Turkey: Erzincan (1992,  $M_w = 6.6$ ), Dinar (1995,  $M_w = 6.2$ ), and Duzce (1999,  $M_w = 7.2$ ). For each building, five structural parameters have been assigned. Four different levels of damage have been originally defined (None, Light, Moderate, and Severe+Collapse). The dataset was analyzed using a machine learning algorithm named CART (Classification And Regression Trees) (Breiman et al. 2017) for different combinations of the structural performance levels.

As mentioned previously, the main purpose of this study is to use a machine learning approach to estimate the observed damages during the 2009 L'Aquila event and defines the main predictive parameters influencing the structural damage. We consider a set of 3060 buildings, reduced to 2532 after the cleaning procedure, characterized by the 21 features displayed in Tables 1 and 2. Most of these structures are located in the historical city centers. The target variable is the level of damage, assigned as light, moderate, and severe. The key difference of this study from the previous ones is that our dataset contains a significantly higher and uniformly distributed number of buildings classified as light, moderate, and severely damaged which reduces the sampling bias in the prediction models. Moreover, each building is extraordinarily well-described through 21 characteristics. This allows us to extensively take into account the complexity of the seismic event, building characteristics, and the local site effects.

The paper is organized as follows: we initially describe the data acquisition and preparation procedure followed by a theoretical introduction to the machine learning tools. Then, we present and discuss the performance of classifiers and compare the results obtained using the height of the building provided by Regione-Abruzzo (2019) with that obtained through aero-photogrammetric techniques. Finally, we summarize the main findings and describe the possible developments of this work.

## 2 Dataset preparation and preliminary analysis

### 2.1 Dataset preparation

The data considered in this study come from different sources and have different formats, such as raster (mainly GeoTiff), vector (shapefile, geojason), and comma-separated values (.csv). The dataset construction represents a crucial part of our workflow and is carefully described. The choice of sources was made taking into consideration that the data would be processed with GIS software (QGIS 3.1 and later <https://www.qgis.org/en/site/>). It follows that all the data had to be either already georeferenced or structured in an appropriate form to be georeferenced. Once the suitable sources were chosen, it was possible to carry out comparisons and calculations between different datasets on a geographical basis, overlay layers from different sources, and create new maps and new datasets.

Some data matrices collected during the project *Open Data Ricostruzione* (GSSI 2019b), despite representing the characteristics of buildings affected by the earthquake, are not georeferenced. These structural characteristics are essential to the ML tools, such as the severity of the damage, the type of construction, the number of structural units, and the year of construction. Such matrices are constructed by inserting a progressive number, typically an ID, which in this case is not comparable with the ID of the shapefile matrix downloaded from the *Abruzzo Region cartographic portal* (Regione-Abruzzo 2019) representing the buildings. It was therefore necessary to identify a field of the matrix that could be uniquely superimposed on a georeferenced datum. This datum was identified in another georeferenced dataset containing field values, specifically a column of unique data, in common with that essential to the ML process, thus allowing georeferencing. To perform the union of these datasets, the native QGIS plugin “*Join attributes by field value*” was used. The data used for the georeferencing, observed with the QGIS software, are made up of polygons representing the footprint on the ground of the buildings subjected to the post-earthquake reconstruction process. Such polygons, called *aggregates*, while not showing any internal subdivision, can contain a single structural unit or more. Very often the structural units forming the aggregate have different characteristics, such as height, construction type, and restructuring phases carried out at different times. In the dataset used for georeferencing, the distinction between structural units of an aggregate is not made and when a distinction exists, it is on the basis of different parameters, such as building owners.

From the point of view of evaluating the response of buildings to seismic actions, this distinction is fundamental, providing more detailed dimensional information such as area, height, and the number of vertices of the building. In the shapefile downloaded from the *Abruzzo Region cartographic portal*, the distinction of the structural units making up an aggregate is instead visible by observing the polygons with the QGIS software Fig. 3. For the reasons mentioned above, further merging between the data arrays was performed. Specifically, the matrix obtained georeferencing the data from the *Open Data Ricostruzione* database, was combined with the matrix provided by the *Abruzzo Region cartographic portal*. The resulting polygon shapefile is a georeferenced matrix containing the distinction of the structural units and many other parameters useful for the ML computational process.

The advantage of having georeferenced data lies in the possibility of overlapping different layers, be they raster or vector maps, to be able to build new datasets deriving from their combination or to implement an existing one with new data. This advantage was exploited to enrich the last shapefile described above, with the data of the web portal <http://shakemap.ingv.it/shake4/> (INGV 2020; Michellini et al. 2020). The values of





**Fig. 3** Distinction of the structural units: left side in orange from CTR, right side in green from Open Data Ricostruzione DB

PGA, PGV, SA0.3s, and Vs30 were transferred, using the QGIS software, to a single point inside each polygon. Each point represents a single structural unit, and for each of them, the number of vertices from which it is composed was calculated. These values complete the range of parameters useful for ML analysis. In Tables 1 and 2, all of the features regarding the building characteristics and geophysical aspects are displayed, respectively, along with the corresponding references.

**Table 1** Buildings features

Features	Type
Construction technique (GSSI 2019b)	Categorical
Aggregation type* (GSSI 2019b)	Categorical
Position (GSSI 2019b)	Categorical
Number of units* (GSSI 2019b)	Numerical
Age (GSSI 2019b)	Categorical
Height* (Regione-Abruzzo 2019)	Numerical
Mean area (GSSI 2019b)	Numerical
Total Area* (GSSI 2019b)	Numerical
Number of vertex* (Regione-Abruzzo 2019)	Numerical

**Table 2** Geophysical features

Features	Type
Geographical coordinate (GSSI 2019b)	Numerical
Peak ground acceleration (INGV 2020)	Numerical
Peak ground velocity (INGV 2020)	Numerical
Spectral acceleration at 0.3 (Vs30) (INGV 2020)	Numerical
Shear wave vel. at 30 m(INGV 2020)	numerical
Epicenter distance (Petricca et al. 2021)	Numerical
Depocenter distance (Petricca et al. 2021)	Numerical
St coefficient (GSSI 2019b)	Numerical
Ss coefficient (GSSI 2019b)	Numerical
Acceleration (GSSI 2019b)	Numerical
Slope (GSSI 2019b)	Categorical

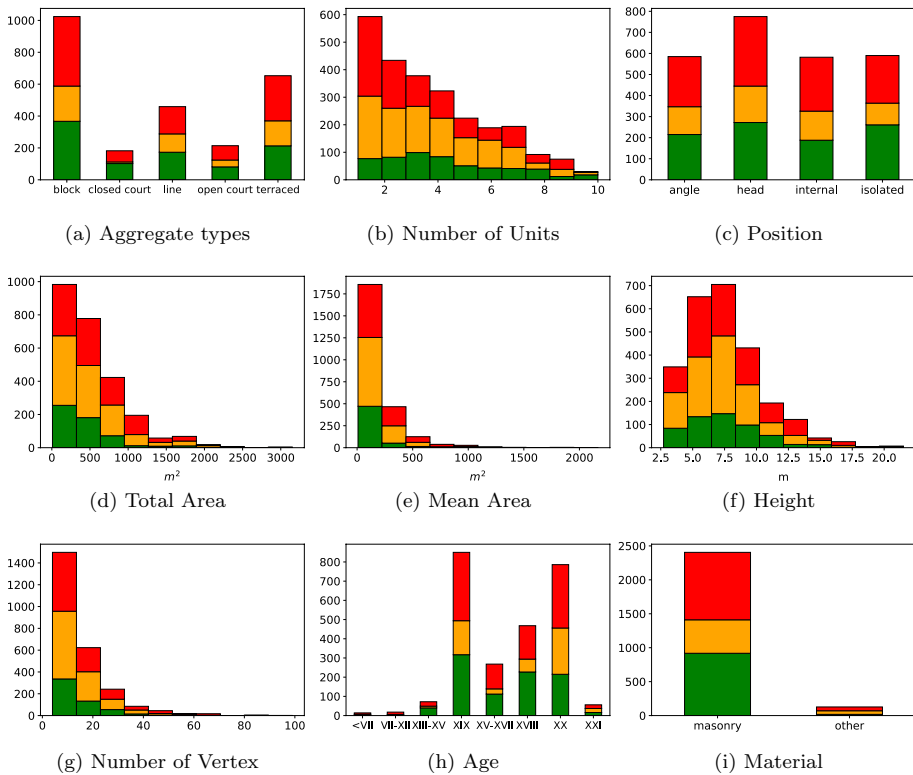
## 2.2 Dataset description

The dataset, in its final form, contains 2532 georeferenced points. Among them, 539 are located in the city of L'Aquila as displayed in Figs. 10 and 11, respectively.

The dataset contains 21 attributes for each building, which can be grouped into two main categories:

- *Buildings features*: construction techniques (such as reinforced concrete, masonry, steel), aggregation type, position (with respect to the aggregate), number of units (in the aggregate), height, surface aggregate area, mean area, number of vertices (in the aggregate), and age.
- *Geophysical features*: geographical coordinate (WGS84/UTM 33 N), distance from the epicenter, distance from the depocenter, peak ground velocity, peak ground acceleration both from the INGV (2020), time-average shear wave velocity to 30 m of depth (Vs30), spectral acceleration at 0.3 s, coefficient of stratigraphic amplification (local site effects), coefficient of topographic amplification, slope, maximum design acceleration value.

The *Buildings features* are summarized in Table 1. The dataset contains many aggregated buildings (here simply called *aggregates*), namely buildings connected to each other by common walls. This configuration is very common in Italy, especially in old town centers, and affects the dynamic response of the system (see for example Stavroulaki 2019 and reference therein). For this reason, it represents, *a priori*, an important parameter for our analysis. As we say before some of the features refer to the *aggregates*, namely height, total area, and the number of vertexes. Also, the number of structural units contained in each *aggregate* is reported. Dividing the total area by the number of units, we get the mean area of the unitary structure. This value corresponds to the real surface area of the building just if the number of units is equal to 1. The kinds of *aggregate* we are considering and the position of the building within the *aggregate* are also reported. Finally, the feature 'Age' assigns the century of construction of the building. For the sake of clarity, all the variables that belong to the *aggregate* and not to the single building are indicated with an asterisk (\*).



**Fig. 4** Building features distribution. Red, orange, and green are heavy, moderate, and light levels of damage, respectively

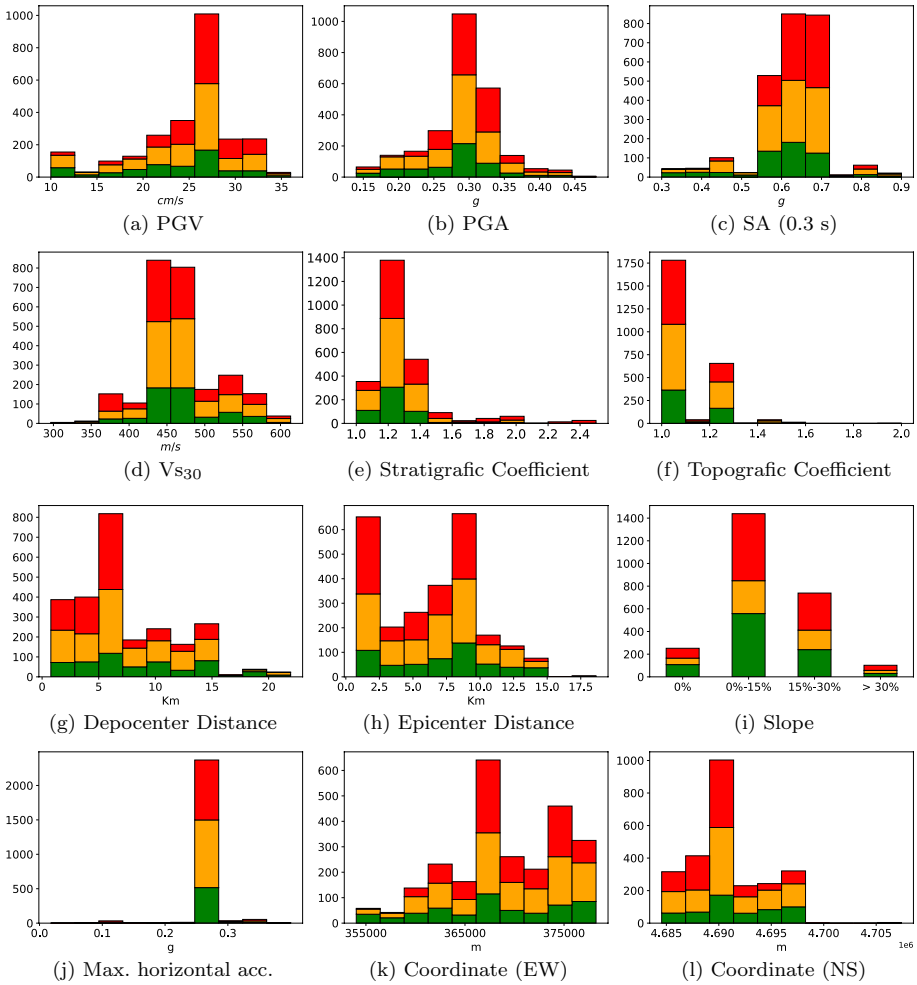
In Table 1, all the *Building features* are listed and the type of data is properly indicated for each feature. The distribution of this subset of predictor variables is shown in Fig. 4.

Within the *geophysical features* (see Table 2 and Fig. 5) peak ground velocity, peak ground acceleration, shear wave velocity at 30 m, and spectral acceleration at 0.3 s have been provided by the platform Shakemap (INGV 2020). We just point out that values of PGV, PGA, and SA at 0.3 are interpolated between the available recording stations.

To account for the building site to seismic source distance, two parameters have been selected: the distance from the epicenter (42.34 N; 13.38 E) and from the depocenter (42.32 N; 13.44 E). While the first one is a commonly used variable, the depocenter needs further explanation. It can be defined as the point that suffers the maximum vertical displacement and it has been observed that the distance from the depocenter could affect the damage distribution (Petricca et al. 2021). Therefore we decide to include both the depocenter and epicenter in our analysis, at least at this initial stage, together with the geographic coordinates.

The coefficient of stratigraphic amplification ( $S_s$ ), topographic amplification ( $S_t$ ), and the maximum horizontal acceleration values ( $a$ ) come from GSSI (2019b). According to the Technical Standard of Italian Constructions (2018), the coefficient  $S_s$  is assigned according to the subsoil properties, whereas the coefficient  $S_t$  depends on the local topography. The acceleration  $a$  is the maximum design horizontal acceleration.





**Fig. 5** Geophysical features distribution. Red, orange, and green are heavy, moderate, and light levels of damage, respectively

Some (less than 5%) stratigraphic amplification coefficient and acceleration values were clearly out of scale. They have been corrected manually, assuming a typo or, more likely, a conversion problem between one format and another in the various phases of the construction of the dataset. We specify that the PGV, the PGA, SA at 0.3 s, and the distance from depocenter and epicenter are the variables that, in this study, refer to the event of the 2009 seismic event in particular.

The spatial distribution of PGA, PGV, SA<sub>0.3</sub> and V<sub>s30</sub> are displayed in Figs. 6, 7, 8 and 9.

The target feature is the level of damage. Six levels of damage have been, originally, assigned to each building as follows (with the number of buildings in each level in parenthesis):

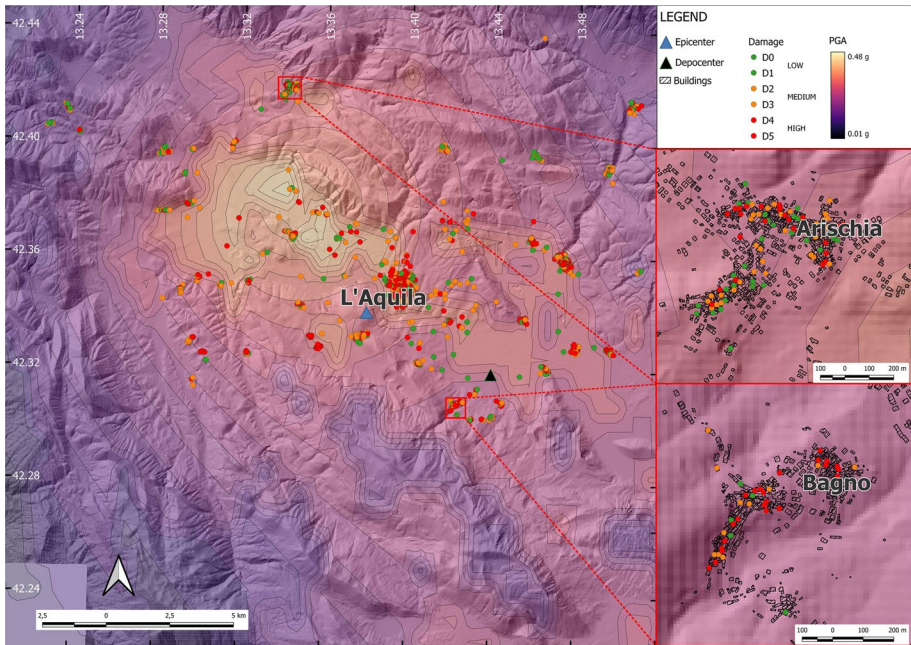


Fig. 6 Spatial distribution of peak ground acceleration (INGV 2020)

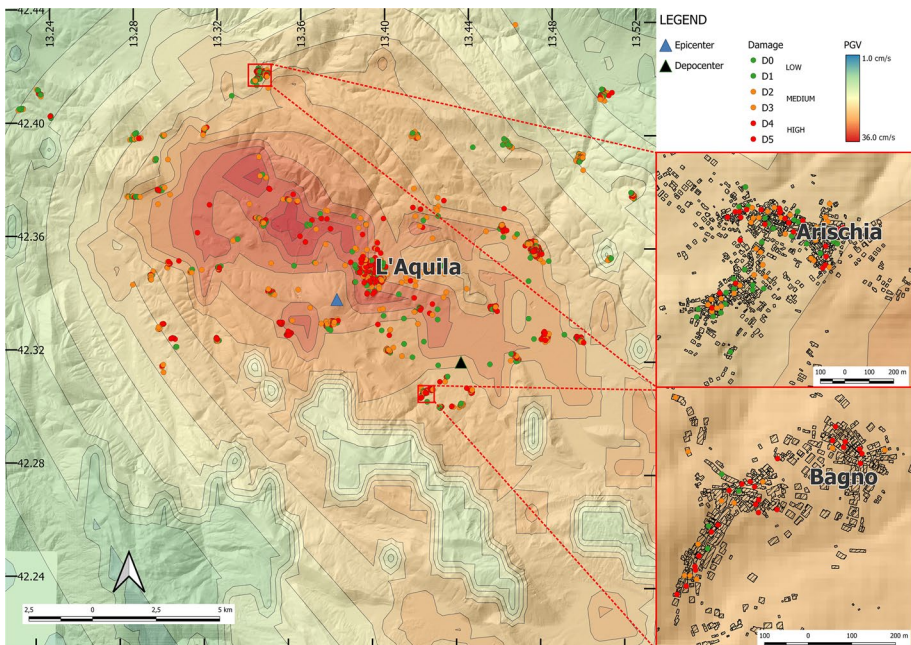


Fig. 7 Spatial distribution of peak ground velocity (INGV 2020)



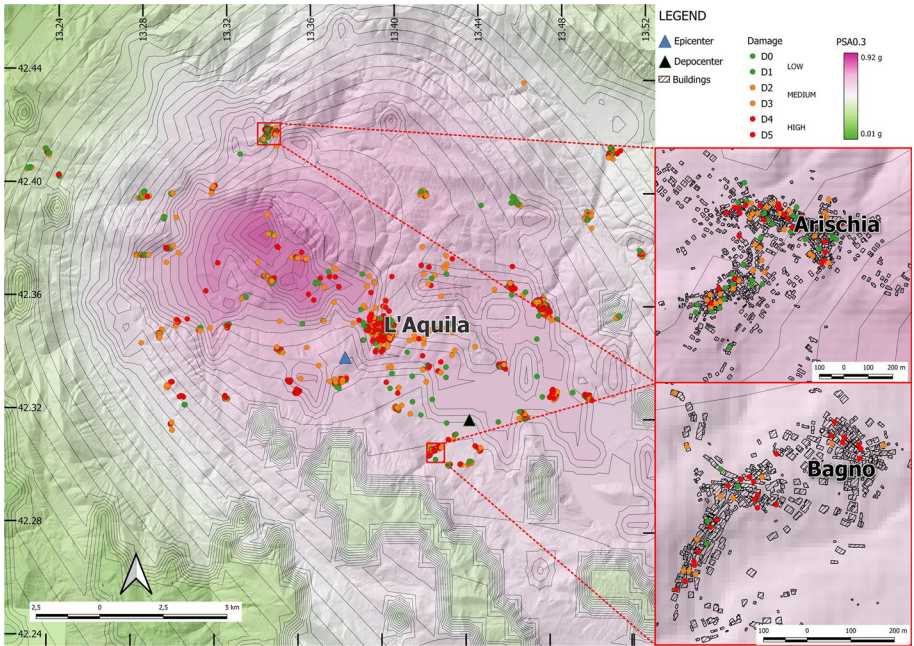


Fig. 8 Spatial distribution of spectral acceleration at 0.3 s (INGV 2020)

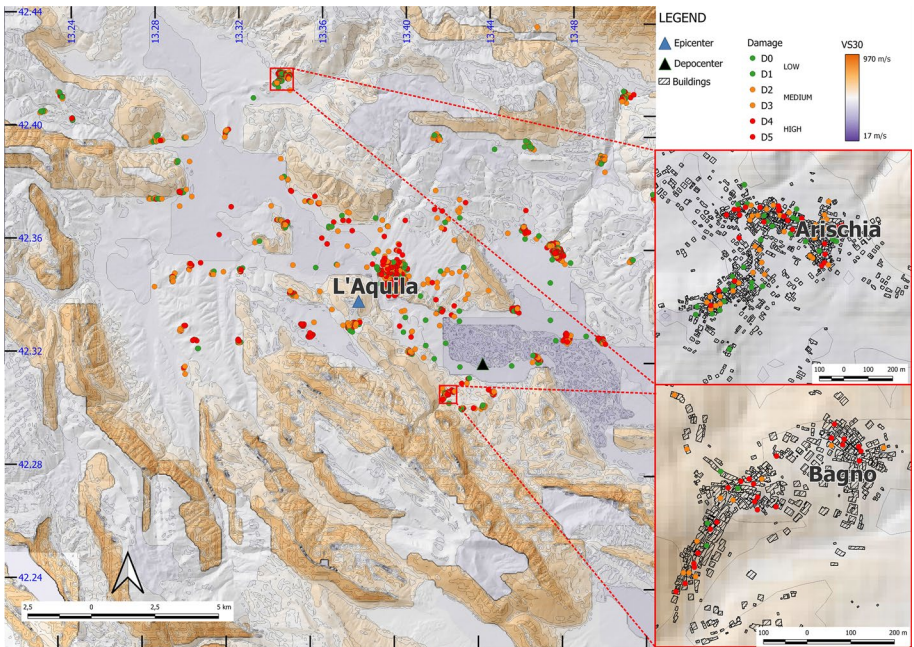


Fig. 9 Spatial distribution of the time-averaged shear wave velocity for the top 30 m (INGV 2020)

- D0: no damage (73).
- D1: light damage (473).
- D2: moderate damage (511).
- D3: medium damage (539).
- D4: serious damage (429).
- D5: heavy damage or collapse (507).

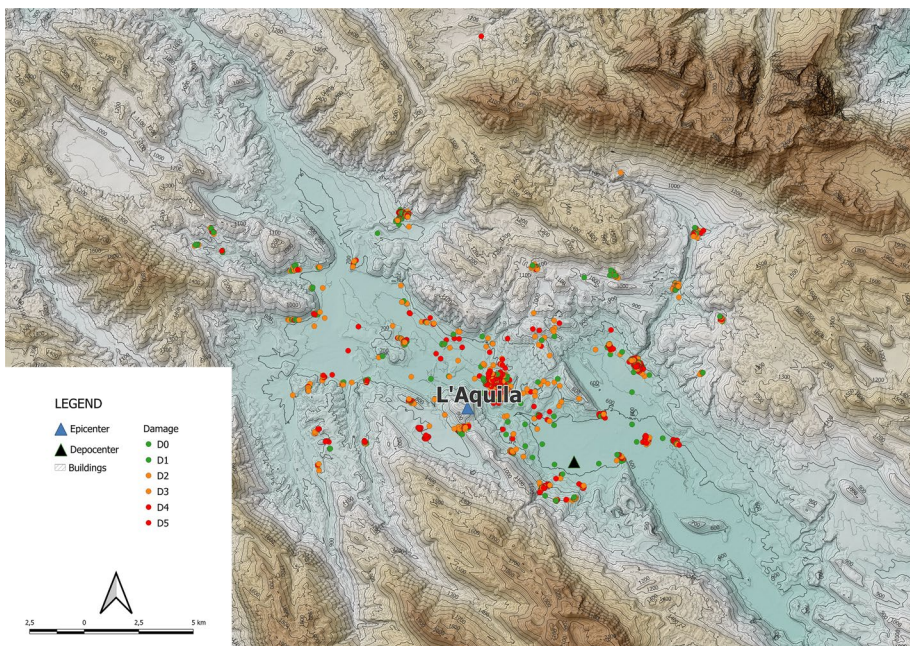
However, due to the dimension of the dataset, in order to improve the accuracy of ML tools, we consider just three levels, namely light, moderate, and heavy damage, as follows

- D0–D1 light damage (546).
- D2–D3 moderate damage (1050).
- D4–D5 heavy damage (936).

The spatial distribution of all the buildings and of those contained within the city walls are displayed in Figs. 10 and 11, respectively.

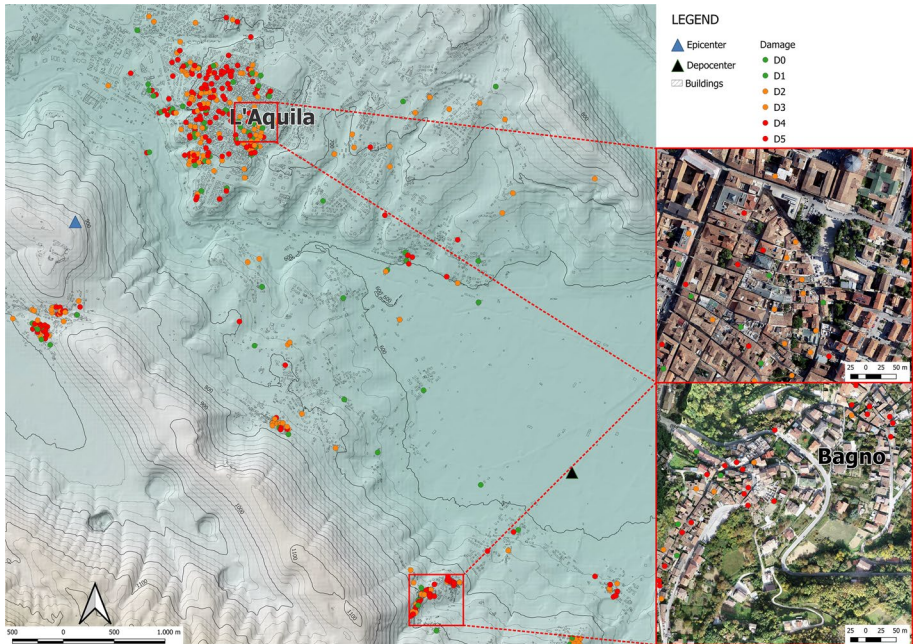
### 3 ML tools

In this section, we briefly describe the ML tools used to analyze our dataset. The first subsection is dedicated to the algorithms for feature selection, the second one concerns the supervised ML tools, and the last is the hyperparameters optimization.



**Fig. 10** Spatial distribution of the buildings located in the city center of L'Aquila. In green, orange, and red the building reported light, moderate, and heavy levels of damage, respectively





**Fig. 11** Spatial distribution of the 2532 buildings included in the dataset. In green (D0–D1), orange (D2–D3), and red (D4–D5) are the building reported light, (546), moderate (1050), and heavy (936) damaged levels, respectively

### 3.1 Features selection

Our dataset contains 21 features (plus the target variable). All of them could be *a priori* useful, but we do not expect that all of them equally contribute to the target variable prediction. Therefore, before starting the data analysis procedure, we apply a feature selection technique to reduce the number of variables. This step is often mandatory for a good data analysis workflow. It allows us to reduce the computational costs and improve the predictive power of ML tools, by eliminating the noncorrelated and redundant features.

A strategy commonly used to reduce the dimension of the dataset is the variable reduction techniques, such as principal component analysis (PCA) and linear discriminant analysis (LDA). However, we avoid this approach because we want to preserve the interpretability of the results, and we prefer to employ the feature selection techniques that can be grouped into (Alelyani et al. 2018; Liu and Yu 2005; Solorio-Fernández et al. 2022)

- Filter methods (FM).
- Wrapper methods (WM).
- Embedded methods (EM).

Since our dataset contains mixed data partially numerical and partially categorical, not all algorithms can be successfully used. The problem has recently been addressed in detail in Solorio-Fernández et al. (2022), where some of the most used algorithms for the feature selection of mixed datasets are reported and compared.



It is not the purpose of this article to offer an exhaustive review of feature selection techniques; we will try to summarize the pros and cons of each approach in order to justify the choice of the algorithm used in this paper.

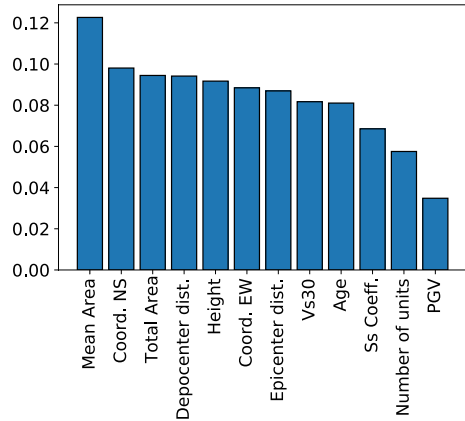
The FM can be based on univariate or multivariate statistics. In the univariate approach, the characteristics are considered independently from each other; therefore, the algorithms are not able to identify the redundant characteristics, because the feature is never compared. These are very efficient algorithms from the computational point also due to their height scalability. The multivariate-based algorithms are slightly more expensive from a computational point of view but are able to identify the redundant and irrelevant characteristics because the importance of each feature with respect to the others is evaluated. Examples of FM are variance threshold (VT) and correlation feature selection (CFS). FMs are usually quite fast but not always exhaustive and therefore are usually used in the pre-processing steps especially if the number of features is particularly high, that is not the case with our dataset.

Wrapper methods are computationally expensive but usually provide better performances in terms of accuracy and overfitting prevention. They use different subsets of characteristics to identify the features having the best performance according to an evaluation criterion. Belong to this group, the *Boruta algorithm* (R Core Team 2021; Kursa and Rudnicki 2010) and *Recursive Feature Elimination*, that will be employed within this paper. Hybrid methods usually combine the filter and wrapper approach in the same algorithm; they are less expensive from the computational point of view but also less accurate compared to the WM. Since the dataset in question has a moderate size and there is no objective computational cost problem, it seems reasonable to use wrapper-type models for the selection of characteristics. Before selecting the best wrapper algorithm to employ, we pre-analyze our dataset using four different classifiers that are Random Forest (RF), Decision Trees (DT), K-nearest neighbors, and Linear discriminant analysis (LDA). For each of them, the default configuration of the hyperparameters has been selected, and then each algorithm has been trained on the training subset containing the 70% of the data points and tested on the remaining part. The best accuracy has been obtained for using the RF which correctly classifies the of 57% the data points in the test set. It, therefore, seems reasonable to use a random forest-based algorithm for the feature selection. The worst performance was instead obtained from LDA, 0.48 of accuracy, which therefore was not considered in the supervised analysis of the dataset, described in the next section.

We decide to employ the *Boruta algorithm* that has been originally developed in R (R Core Team 2021; Kursa and Rudnicki 2010) and more recently translated to Python. In summary, it evaluates the importance of each feature by creating a shadow dataset obtained from the original one, just scrambling the values within each column. Then the two datasets (original and shadow) are analyzed by an RF classifier. In this way, we identify the features of the original dataset that most influence the classifier's predictive capacity. The process is iterated many times until all redundant features are removed. The 12 features selected through the Boruta algorithm are: Mean Area, Total Area\*, Number of Units\*, Distance from the Depocenter, Distance from the Epicenter, Height\*, Coordinate (EW direction), Coordinate (NS direction), PGV, Vs30, Ss coefficient, and Age.

It is interesting to observe that among these, two refer to the building space position (geographic coordinates), six refer to the building itself or to the ground on which it is built (Mean Area, Total Area, Number of Units, Ss coefficient, Vs30, Age and Height) and three (PGV, distance from the epicenter and from the depocenter) to the specific event we are considering. These initial observations immediately suggest that adequate seismic planning must take into account the following main features: the potential events that may occur

**Fig. 12** Feature importance obtained using the RF algorithm on the 12 features selected by the Boruta algorithm



in the area of interest, the local site effects, and the physical properties of the buildings of interest. These findings are consistent with both the previous observations on seismic effects and the main seismic design criteria in modern building codes. In Fig. 12, the score for each of the 12 features obtained with the RF algorithm, is reported. According to our analyses, the characteristic that most influences the level of damage is the average area, that is the total area of the aggregate divided by the number of buildings contained in it. Both of these characteristics (total area and number of units) are present among the 12 features selected by the Boruta algorithm, but it is their combination that has a greater impact on our analysis. Also, the distance of the depocenter (rather than that from the epicenter) has a strong effect on the damage as already observed in Petricca et al. (2021).

### 3.2 Supervised ML tools

We consider three different classifiers in this study: the  $k$ -nearest neighbors ( $k$ -NN), the random forest (RF), and the decision trees (DT). We do not account for linear discriminant analysis, the tools proposed for example by Mangalathu et al. (2020), that is not able to capture the strictly nonlinear nature of the phenomenon in our study. Furthermore, other approaches, such as the convolution neural network, have been effectively implemented in the literature, especially for the analysis of satellite and UAV data (Ci et al. 2019; Fujita et al. 2017), and are of potential interest in future studies. However, in this first work, we have chosen the most practical and validated technique which is supervised ML.

Next, we briefly review the tools employed within the paper followed by the results obtained with each technique. We note that all the machine learning models have been implemented herein using Scikit-learn, a Python tool specifically designed for data analysis. We refer to Raschka and Mirjalili (2017) for a very useful and complete guide to Scikit-learn and to the most commonly used machine learning tools.

### 3.3 K-nearest neighbor (KNN) algorithm

The *KNN algorithm*, first introduced by Fix and Hodges (1951), belongs to the class of the nonparametric models and is one of the simplest and most used algorithms for classification and regression. It assigns the label to each element according to the characteristics of

the ' $k$ ' elements of the train subset 'closest' to the element (each data point) under analysis. Its functioning can be summarized as follows:

**Step 1** The first step is to select an appropriate metric for the distances. The choice of metric depends on the characteristic of the dataset we are using and many different options are available in Scikit-learn. The most common choice is the Euclidean metric. In this case, the distance between two elements  $\mathbf{x}^i$  and  $\mathbf{x}^j$ , belonging to  $\mathbf{R}^n$ , reads

$$d(\mathbf{x}^i, \mathbf{x}^j) = \sqrt{\sum_{n=1}^N (x^i(n) - x^j(n))^2}$$

**Step 2** Using the metric selected above, we identify the  $k$  points of the training dataset that are closest to each element that must be classified.

**Step 3** Finally, we assign the label using the principle of majority vote. If two labels reach the same score, then the algorithm will take into account the distance and will assign the most frequent label to the closest neighbors (Raschka and Mirjalili 2017).

### 3.4 Decision trees

*Decision tree* is a very useful classifier for many purposes because of its intrinsic interpretability (Raschka and Mirjalili 2017). The algorithm can be shortly described through the following two steps:

**Step 1** Initially, we fix the Information Gain (IG), which can be seen as the difference of the impurity between the parent node  $I_p$  and the  $M$  child node  $I_i$ , namely

$$\text{IG}(D_p, g) = I_p - \sum_{i=1}^M \frac{N_i}{N_p} I_i$$

where  $N_i$  and  $N_p$  are the number of data points for the  $i$ th child node and for the parent node, respectively. Impurity can be assigned in many different ways; the most common are *Gini Impurity*, *Entropy*, and *Classification Error*. Then, starting from the root we split our dataset to maximize IG.

**Step 2** The process will be iterated until the maximum depth value  $k$ , fixed *a priori*, has been reached.

### 3.5 Random forest

The *Random Forest algorithm* is one of the most useful ML tools (Liaw and Wiener 2002). In summary, it can be interpreted as an ensemble of decision trees (Raschka and Mirjalili 2017), and works as follows:

**Step 1** We start from a random bootstrap sample of size  $n$  and grow a decision tree.

**Step 2** We iterate the procedure above  $k$  times. Each of the iterations provides a result, namely a value for the target variable.

**Step 3** Finally, we assign the label using the majority vote algorithm.

### 3.6 Hyper-parameters optimization

In this subsection, we will briefly discuss how to optimize the hyperparameters of a classifier to achieve optimal performance.

Parameter optimization can basically be done in three different ways:

- *Manual optimization*: the analyst selects some values of the hyperparameters based on his experience and knowledge of the dataset and modifies them manually until reaching optimal accuracy.
- *Grid search*: the analyst selects a grid of values for each hyperparameter. The model is trained and evaluated on each combination of the values entered in the grids. This is a very accurate method if dense grids are used, but it has a very high computational cost.
- *Random search*: also in this case a grid or a range of values is assigned for each hyperparameter and the number  $n$  of combinations to consider. The system randomly selects and evaluates  $n$  combinations of hyperparameters. This is a computationally efficient method that usually best approximates the optimal combination of hyperparameters.

Within this paper, we employ the Scikit-learn function `RandomizedSearchCV`, which combines the Randomized Search approach described above and the cross-validation technique to control the overfitting (see Raschka and Mirjalili 2017 for more detail). Each model was optimized using the training dataset, over a space of 200 hyperparameter combinations.

## 4 Results and discussion

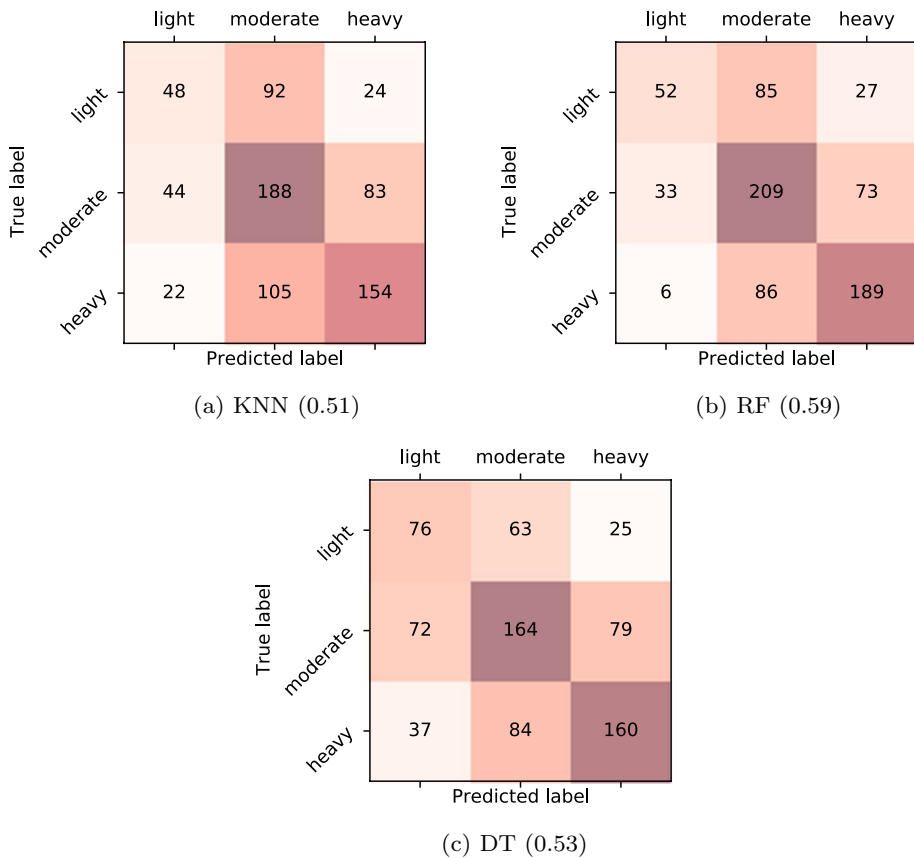
We present and analyze the results of damage state predictions using the datasets and machine learning tools described above.

Of the 2532 data points that constitute the database, 2278 were used to train the models and 760 (30%) to test them, as in Mangalathu and Jeon (2018). The distribution of the target variable, within the train and test sets, follows that of the original dataset, as displayed in Table 3.

We consider two different datasets, named  $S_{21}$  and  $S_{12}$ , containing the same number of data points (2532), but a different number of predictor variables. The first is the complete dataset that includes all of the mentioned 21 features, whereas in the second one there are just the 12 variables selected by the Boruta algorithm. To compare the three classifiers (and the two datasets), we use the confusion matrix, which allows quick visualization of the distribution of the predicted tags (damage states) versus the originally assigned (observed) ones. The main diagonal contains the number of buildings of which the damage state is correctly classified, while the numbers in off-diagonal elements indicate the buildings with incorrectly classified damage states.

**Table 3** Data distribution

Damage	Total	Train	Test	%
Light	546	382	164	22
Moderate	1050	735	315	41
Heavy	936	655	281	37
Tot	2532	1772	760	100



**Fig. 13** Performance of the three classifiers using dataset  $S_{21}$ . In brackets is the accuracy of each method

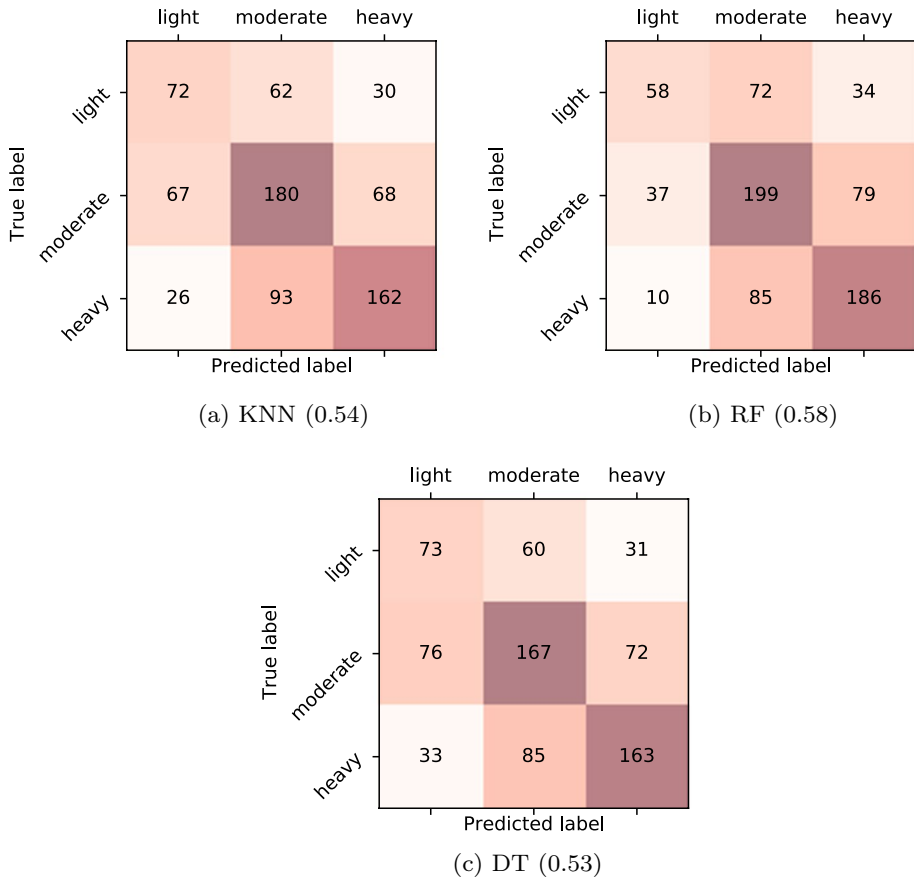
In Figs. 13 and 14, the scores are obtained employing the dataset  $S_{21}$  and  $S_{12}$ , respectively. Under each map, the overall accuracy of the method is also shown. It represents the overall percentage of correctly classified buildings and can be defined as:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \quad (1)$$

where TP, FP, TN, and FN stand for true positives, false positives, true negatives, and false negatives, respectively. The performance of the two datasets is comparable, although the reduced one yields slightly higher accuracy at least for the KNN classifier.

This confirms that a reduction in the number of variables, and consequently the dataset complexity, allows for improvement in the tool performance while lowering the computational cost. For both datasets, the RF algorithm provides higher accuracy. It is thus important to decide and eliminate less critical structural or geophysical explanatory variables before a prediction analysis. The precision is the ratio between the number of tags assigned correctly and the total number of times that the specific tag has been assigned by the model including false positives. Conversely, recall is the ratio between the number of tags assigned correctly and the total number of times that the specific tag appears in the





**Fig. 14** Performance of the three classifiers using dataset  $S_{12}$ . In brackets is the accuracy of each method

dataset including false negatives. High values of recall and precision values correspond to a high predictive capacity of the model.

From Figs. 13b and 14b, it is clear that the RF algorithm has an excellent ability to distinguish between the most seriously damaged buildings from those that have suffered only minor damage. In fact, for the reduced dataset we can see that only 10 ‘red’ tagged buildings (the 3% out of a total of 281) were classified as ‘light damage.’ Indeed for the red tags, the precision and recall are quite high, reaching 62 and 65%, respectively. This is anticipated since heavy damage and light damage correspond to very different numerical values of the explanatory variables. This observation is consistent with the findings of a past study using discriminant analysis to predict damage states of buildings after large earthquakes in Turkey (Askan and Yucemen 2010). The results of the entire dataset are slightly better in this case: only 6 ‘red’ tagged buildings were classified as ‘light damage.’ In this case, the precision and recall reach 65 and 67%, respectively. As a general remark, we observe that the maximum accuracy achieved by the RF algorithm is around 65%, although in line with many of the results available in the literature, is probably improvable using a more well-constrained dataset. However in this work, we preferred to preserve the entire dataset even though the values of some features are poorly represented within the dataset, and this has

certainly worsened the overall performance of our classifiers. From now on we focus on the reduced dates; however, similar conclusions can be drawn by looking at the results of the complete dataset. Observing Fig. 14b we can see how the performance for the ‘orange’ buildings is also quite good with recall and precision equal to 63 and 55%, respectively. Among the total of moderate 315 data points analyzed in the train set, approximately 10% of them were classified as light damage. The results for light damage (green) buildings are not satisfactory, probably because they are less numerous in the dataset. Less than 40% buildings with minor damage were classified correctly, with a recall of 30%. Slightly worse is the performance of the DT and KNN classifiers which reaches an accuracy of 0.55 and 0.54, respectively.

In compiling a damage inventory after earthquakes, the exact height of the buildings could be unknown. In this case, aereophotogrammetry is the simplest and most economical tool to get this information. In addition, it is often more accurate data that also takes into account the slope of the ground and the presence of towers or bell towers. To study the significance of the accuracy of building heights, we introduce another dataset where the height of the buildings is given by aereophotogrammetric techniques ( $S_{12UAV}$ ). It will be compared to a subset of  $S_{12}$  with the same number of buildings as  $S_{12UAV}$  but heights provided by regional technical maps (Regione-Abruzzo 2019). From now on we call this dataset  $S_{12RTM}$ . We note that both  $S_{12UAV}$  and  $S_{12RTM}$  contain 12 features, those selected by the Boruta algorithm.

Before presenting the data analysis, we briefly illustrate the process by which we calculated the height of the buildings.

From the DEMs it was possible to extract the Digital Terrain Models (DTMs) and the Digital Surface Models (DSM), the former represents the surface of the soil without elevations (buildings, trees, vehicles), the latter the elevations without the soil obtained subtracting the DTMs to the DEMs. It follows that DEMs and DTMs provide elevation values with respect to the sea level, while the DSMs with respect to the ground (Fig. 15a–c). Through the analysis of the DSM raster, the QGIS software was forced to perform a statistical analysis of the heights, confining the calculation inside each polygon that represents a single structural unit. The maximum, minimum, and average heights were thus calculated, although only the maximum has been used for our analysis. These values were then transferred to the data matrix useful for the ML calculation. A total of about 180,000 aerial georeferenced photographs were acquired, through which dense point clouds, Digital Elevation Models (DEMs), and orthomosaics were produced with photogrammetric processing. The DEMs of the built areas have a very high resolution, about 6 cm/pixel, and are the first ever produced for the city of L’Aquila at such a resolution.

The obtained dataset contains 1457 buildings, of which 656 (45%) have been cataloged as red buildings (i.e., seriously damaged), 565 (39%) with moderate damage, and 236 (16%) with light or no damage (vs the 22% of the datasets analyzed above). The same distribution of target variables is maintained for training and testing sets. For both datasets, the accuracy is in general smaller than those obtained for  $S_{12}$ , due to the drastic reduction in the number of data points.

In this case, the best performance is obtained from the RF classifier; indeed, it shows a total accuracy of 57% for  $S_{12RTM}$ , against 55%. In general, from Figs. 16 and 17, we can conclude that the two dataset provides comparable results. This finding justifies the use of aereophotogrammetric techniques to evaluate the heights of buildings if they are not available from other sources.

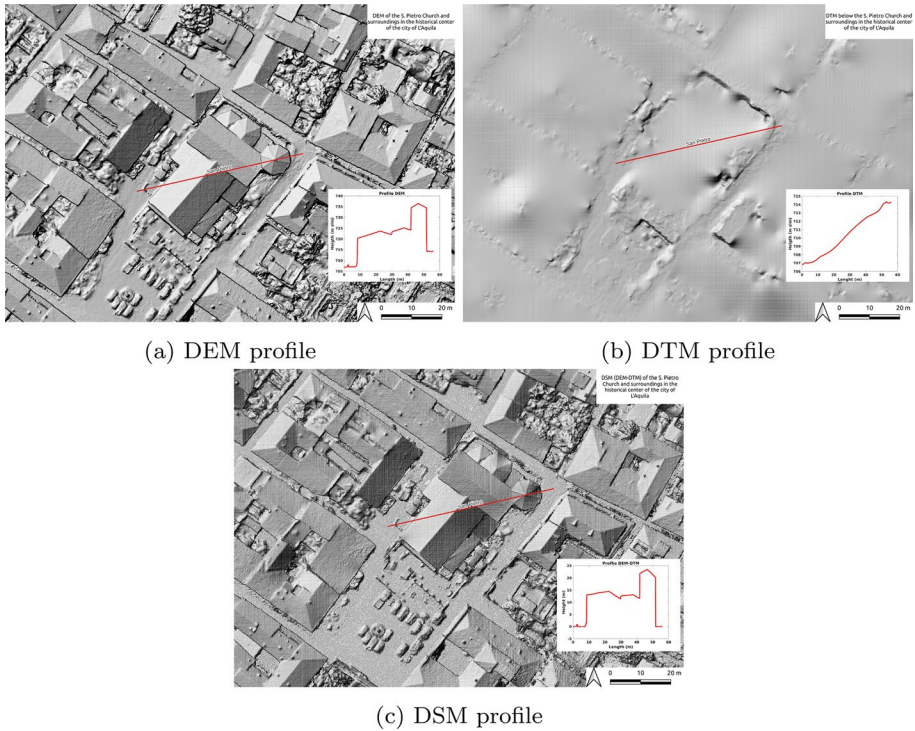
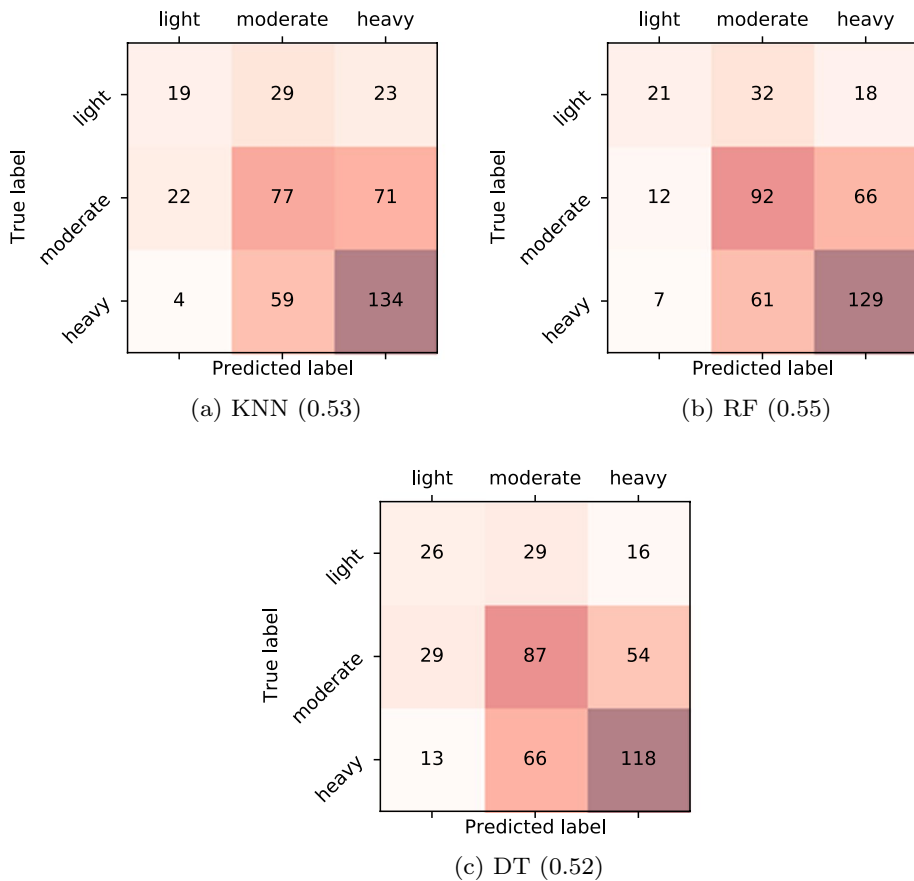


Fig. 15 Workflow for buildings height calculation

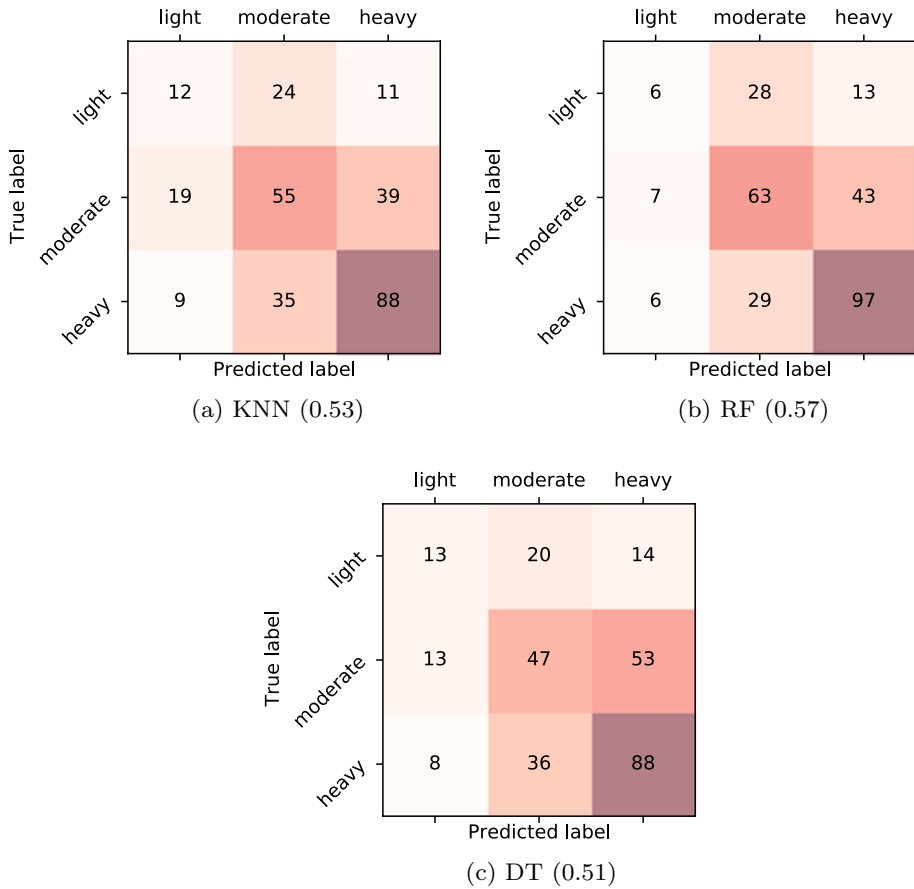
## 5 Conclusions

Damage datasets from the 2009 L'Aquila earthquake are used to apply machine learning algorithms, and the most accurate forecast factors are presented. Although our dataset is not very large, it provides unprecedented detail on the structures and soil in a typical town in the Central Apennines, a high-risk area that has been struck by several recent earthquakes (L'Aquila 2009, Amatrice and Norcia 2016). Our dataset can be used to provide useful insight into the factors that significantly affect how a building reacts during an earthquake. On the other hand, it can serve as a standard against which other machine learning methods and feature selection strategies can be evaluated. When compared to other datasets of a similar size, like the one used in Mangalathu and Jeon (2018), it has a rather even distribution of buildings reported with light, moderate, and heavy damage. Any potential biases toward a particular damage state can be mitigated with the help of such a distribution. Aerial photogrammetric surveys of several of the structures are also available and can be used to supplement the dataset in the future. For the first time, a supervised learning approach is used to forecast the seismic damage states of structures hit by the 2009 L'Aquila event. The techniques and datasets employed here allow us to draw the following findings. Less than 3% of buildings with red tags were misclassified as green, showing that the Random Forest algorithm can effectively distinguish between severely damaged and unharmed structures.



**Fig. 16** Performance of the classifiers using dataset  $S_{12UAV}$ , where the height of the building is obtained by aerophotogrammetric techniques. In brackets is the accuracy of each method

The mean area, calculated by dividing the entire area of the aggregate by the total number of structures, is determined to be the primary characteristic affecting damage. Damage distribution in our dataset is also substantially influenced by geographical location, building height, and distance to the hypocenter (as opposed to the epicenter) (Petricca et al. 2021). In the future, we plan to assess the efficacy of other supervised learning methods, maybe in tandem with ensemble models, which permit the integration of multiple methods into a single model. By using the Boruta approach for selecting the features, we were able to cut down on processing time without sacrificing forecast precision. However, it is fascinating to use various feature reduction and selection approaches to a bigger dataset to better understand the impact of each variable. The topic of this forthcoming study. The machine learning algorithms evaluated here can be used in a systematic damage assessment framework where the input ground motions are not interpolated from observable data but are instead simulated with different methods (Di Michele et al. 2022; Evangelista et al. 2017; Askan et al. 2013; Di Michele et al. 2021). A scenario-based methodology like this could be useful for estimating the costs of future earthquakes.



**Fig. 17** Performance of the classifiers using the dataset  $S_{12RTM}$ , where the height of the buildings has been obtained from Regione-Abruzzo (2019). In brackets is the accuracy of each method

## Appendix

In this section, we report for each dataset the values of the hyperparameters employed in our analyses

$$S_{12}$$

- RF: (criterion = entropy, max\_features = sqrt, n\_estimators = 1604, min\_samples\_split = 10, min\_samples\_leaf = 4, max\_depth = 100)
- KNN: (weights = distance, p = 1, n\_neighbors = 20)
- DT : (random\_state = 100, min\_samples\_split = 2, min\_samples\_leaf = 1, max\_features = 12, max\_depth = 130, criterion = gini)

$$S_{21}$$



- RF: (criterion = entropy, max\_features = sqrt, n\_estimators = 1146, min\_samples\_split = 2, min\_samples\_leaf = 6, max\_depth = 120)
- KNN: (weights = distance, p = 1, n\_neighbors = 38)
- DT : (random\_state = 10, min\_samples\_split = 1, min\_samples\_leaf = 2, max\_features = 20, max\_depth = 39, criterion = entropy)

$S_{UAV}$

- RF: (criterion = entropy, max\_features = sqrt, n\_estimators = 1604, min\_samples\_split = 10, min\_samples\_leaf = 4, max\_depth = 100)
- KNN: (weights = distance, p = 1, n\_neighbors = 20)
- DT : (random\_state = 10, min\_samples\_split = 2, min\_samples\_leaf = 1, max\_features = 12, max\_depth = 130, criterion = gini)

$S_{RTM}$

- RF: (criterion = entropy, max\_features = log2, n\_estimators = 1185, min\_samples\_split = 10, min\_samples\_leaf = 1, max\_depth = 10)
- KNN: (weights = distance, p = 1, n\_neighbors = 34)
- DT : (random\_state = 10, min\_samples\_split = 2, min\_samples\_leaf = 1, max\_features = 12, max\_depth = 10, criterion = entropy)

**Funding** Open access funding provided by Gran Sasso Science Institute - GSSI within the CRUI-CARE Agreement. This work was partially supported by the GSSI “Centre for Urban Informatics and Modelling” (Italian Government—Presidenza Consiglio dei Ministri—CUIM project delibera CIPE n. 70/2017).

## Declarations

**Conflict of interest** The authors declare that they have no conflict of interest.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article’s Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

- Alelyani S, Tang J, Liu H (2018) Feature selection for clustering: a review. *Data Clust* 29–60
- Askan A, Yucemen M (2010) Probabilistic methods for the estimation of potential seismic damage: application to reinforced concrete buildings in Turkey. *Struct Saf* 32(4):262–271
- Askan A, Sisman FN, Ugruhan B (2013) Stochastic strong ground motion simulations in sparsely-monitored regions: a validation and sensitivity study on the 13 March 1992 Erzincan (Turkey) earthquake. *Soil Dyn Earthq Eng* 55:170–181
- Baker JW (2015) Efficient analytical fragility function fitting using dynamic structural analysis. *Earthq Spectra* 31(1):579–599

- Breiman L, Friedman JH, Olshen RA, Stone CJ (2017) Classification and regression trees. Routledge
- Ci T, Liu Z, Wang Y (2019) Assessment of the degree of building damage caused by disaster using convolutional neural networks in combination with ordinal regression. *Remote Sens* 11(23):2858
- De Luca F, Verderame GM, Manfredi G (2015) Analytical versus observational fragilities: the case of Petriano (L'aquila) damage data database. *Bull Earthq Eng* 13(4):1161–1181
- Del Gaudio C, De Martino G, Di Ludovico M, Manfredi G, Protà A, Ricci P, Verderame GM (2019) Empirical fragility curves for masonry buildings after the 2009 L'aquila, Italy, earthquake. *Bull Earthq Eng* 17(11):6301–6330
- Di Michele F, Pera D, May J, Kastelic V, Carafa M, Styahar A, Rubino B, Aloisio R, Marcati P (2021) On the possible use of the not-honoring method to include a real thrust into 3D physical based simulations. In: 2021 21st International conference on computational science and its applications (ICCSA). IEEE, pp 268–275
- Di Michele F, May J, Pera D, Kastelic V, Carafa M, Smerzini C, Mazzieri I, Rubino B, Antonietti PF, Quarteroni A et al (2022) Spectral elements numerical simulation of the 2009 L'aquila earthquake on a detailed reconstructed domain. *Geophys J Int*
- Evangelista L, Del Gaudio S, Smerzini C, d'Onofrio A, Festa G, Iervolino I, Landolfi L, Paolucci R, Santo A, Silvestri F (2017) Physics-based seismic input for engineering applications: a case study in the Aterno river valley, Central Italy. *Bull Earthq Eng* 15(7):2645–2671
- Fix E, Hodges J (1951) Discriminatory analysis. Nonparametric discrimination: Consistency properties. Technical Report 4
- Fujita A, Sakurada K, Imaizumi T, Ito R, Hikosaka S, Nakamura R (2017) Damage detection from aerial images via convolutional neural networks. In: 2017 Fifteenth IAPR international conference on machine vision applications (MVA). IEEE, pp 5–8
- GSSI (2019a) Open data L'aquila. <https://www.opendatalaquila.it/>
- GSSI (2019b) Open data ricostruzione. <https://opendataricostruzione.gssi.it/home>
- Harirchian E, Kumari V, Jadhav K, Rasulzade S, Lahmer T, Raj Das R (2021) A synthesized study based on machine learning approaches for rapid classifying earthquake damage grades to RC buildings. *Appl Sci* 11(16):7540
- INGV (2020) Shakemap. <http://shakemap.ingv.it/shake4/>
- Kirçil MS, Polat Z (2006) Fragility analysis of mid-rise R/C frame buildings. *Eng Struct* 28(9):1335–1345
- Kursa MB, Rudnicki WR et al (2010) Feature selection with the Boruta package. *J Stat Softw* 36(11):1–13
- Lallemant D, Kiremidjian A, Burton H (2015) Statistical procedures for developing earthquake damage fragility curves. *Earthq Eng Struct Dyn* 44(9):1373–1389
- Liaw A, Wiener M et al (2002) Classification and regression by randomForest. *R News* 2(3):18–22
- Liu H, Yu L (2005) Toward integrating feature selection algorithms for classification and clustering. *IEEE Trans Knowl Data Eng* 17(4):491–502
- Mangalathu S, Jeon J-S (2018) Classification of failure mode and prediction of shear strength for reinforced concrete beam-column joints using machine learning techniques. *Eng Struct* 160:85–94
- Mangalathu S, Sun H, Nweke CC, Yi Z, Burton HV (2020) Classifying earthquake damage to buildings using machine learning. *Earthq Spectra* 36(1):183–208
- Michelini A, Faenza L, Lanzano G, Lauciani V, Jozinović D, Puglia R, Luzi L (2020) The new ShakeMap in Italy: progress and advances in the last 10 yr. *Seismol Res Lett* 91(1):317–333
- Petricca P, Bignami C, Doglioni C (2021) The epicentral fingerprint of earthquakes marks the coseismically activated crustal volume. *Earth Sci Rev* 103667
- R Core Team (2021) R: a language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. <https://www.R-project.org/>
- Raschka S, Mirjalili V (2017) Python Machine Learning: Machine Learning and Deep Learning with Python
- Regione-Abruzzo (2019) Geoportale. <http://geoportale.regione.abruzzo.it/Cartanet>
- Roeslin S, Ma Q, Juárez-García H, Gómez-Bernal A, Wicker J, Wotherspoon L (2020) A machine learning damage prediction model for the 2017 Puebla-Morelos, Mexico, earthquake. *Earthq Spectra* 36(2\_suppl):314–339
- Rosti A, Rota M, Penna A (2018) Damage classification and derivation of damage probability matrices from L'aquila (2009) post-earthquake survey data. *Bull Earthq Eng* 16(9):3687–3720
- Rovida A, Locati M, Camassi R, Lolli B, Gasperini P (2020) The Italian earthquake catalogue CPTI15. *Bull Earthq Eng* 18(7):2953–2984
- Rovida A, Locati M, Camassi R, Lolli B, Gasperini P, Antonucci A (2022) Catalogo parametrico dei terremoti italiani (cpti15). Versione 4.0. Istituto Nazionale di Geofisica e Vulcanologia (INGV). Italy
- Shinozuka M, Feng MQ, Lee J, Naganuma T (2000) Statistical analysis of fragility curves. *J Eng Mech* 126(12):1224–1231

- 
- Solorio-Fernández S, Carrasco-Ochoa JA, Martínez-Trinidad JF (2022) A survey on feature selection methods for mixed data. *Artif Intell Rev* 55(4):2821–2846
- Stavroulaki ME (2019) Dynamic behavior of aggregated buildings with different floor systems and their finite element modeling. *Front Built Environ* 5:138
- Ugurhan B, Askan A, Akinci A, Malagnini L (2012) Strong-ground-motion simulation of the 6 April 2009 L'aquila, Italy, earthquake. *Bull Seismol Soc Am* 102(4):1429–1445
- Yerlikaya-Özkurt F, Askan A (2020) Prediction of potential seismic damage using classification and regression trees: a case study on earthquake damage databases from Turkey. *Nat Hazards* 103(3):3163–3180

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.