Earthquake Prediction in California Using Regression Algorithms and Cloud-based Big Data Infrastructure

Article *in* Computers & Geosciences · November 2017

DOI: 10.1016/j.cageo.2017.10.011

CITATIONS
105

READS
5,659

4 authors:

Gualberto Asencio Cortés
University of Pablo de Olavide
78 PUBLICATIONS **1,306** CITATIONS
SEE PROFILE

Antonio Morales-Esteban
University of Seville
99 PUBLICATIONS **1,832** CITATIONS
SEE PROFILE

Xueyi Shang
35 PUBLICATIONS **802** CITATIONS
SEE PROFILE

Francisco Martínez-Álvarez
Pablo de Olavide University
209 PUBLICATIONS **6,353** CITATIONS
SEE PROFILE

Earthquake Prediction in California Using Regression Algorithms and Cloud-based Big Data Infrastructure

G. Asencio–Cortés[a], A. Morales–Esteban[b], X. Shang[c], F. Martínez–Álvarez[a]

[a] *Division of Computer Science, Pablo de Olavide University of Seville, Spain – {guaasecor,fmaralv}@upo.es*
[b] *Department of Building Structures and Geotechnical Engineering, University of Seville, Spain – ame@us.es*
[c] *School of Resources and Safety Engineering, Central South University, China – shangxueyi@csu.edu.cn*

Abstract

Earthquake magnitude prediction is a challenging problem that has been widely studied during the last decades. Statistical, geophysical and machine learning approaches can be found in literature, with no particularly satisfactory results. In recent years, powerful computational techniques to analyze big data have emerged, making possible the analysis of massive datasets. These new methods make use of physical resources like cloud based architectures. California is known for being one of the regions with highest seismic activity in the world and many data are available. In this work, the use of several regression algorithms combined with ensemble learning is explored in the context of big data (1 GB catalog is used), in order to predict earthquakes magnitude within the next seven days. Apache Spark framework, H_2O library in R language and Amazon cloud infrastructure were been used, reporting very promising results.

Key words: Earthquake prediction, big data analytics, cluster computing, regression, ensemble learning

1. Introduction

Modern societies are threatened by natural risks and demand a proper preparation to reduce their impact. During the last years many initiatives have merged from modern societies in order to minimize the economical and human impact of natural disasters.

Natural risk is a concept embedded in the collective consciousness of modern societies. Against expected, an objective and universal definition of risk is yet to be established [1]. Nevertheless, it can be defined as a measure of the combined likelihood of occurrence of a threatening event and its potential consequences.

Natural disasters occur when a probable hazard turns into a real event. Then, potential consequences become real human and economic losses. Among natural disasters such as earthquakes, tsunamis, volcanic eruptions, hurricanes, tornadoes, floods and others, earthquakes stand out due to their devastating effects [2]. Earthquakes arrive suddenly and can destroy a whole city or region within seconds causing lost of lives or injures, property damage, social and economic breaks or environmental damage [3]. Moreover, many populated areas stand on seismic zones. Besides, earthquakes can produce correlated effects such as tsunamis [4], landslides [5] and liquefaction [6].

Seismic risk is a combination of seismic hazard and seismic vulnerability [7]. On the one hand, seismic hazard represents a potentially damaging seismic event that can cause damage. On the other hand, the potential consequences are the existing vulnerabilities that show the susceptibility to the damaging effect of the hazard.

Big data analytics has emerged as a very powerful technique. It is typically used to examine huge datasets in order to extract useful information and discover patterns [8]. When such big datasets must be analyzed, computational resources increase and traditional machine learning algorithms require new parallelized implementations that must be launched in clusters [9].

For all the aforementioned, there is a worldwide trend to enhance our understanding of earthquakes in order to increase our ability to manage them [10]. In this paper, earthquake prediction in one of the most seismic and populated areas of the world -California- is explored. So far, standard machine learning

algorithms were been applied to earthquake prediction. However, studied datasets' sizes were typically no bigger than several MB [11]. For this purpose, a 1 GB catalog was been generated, retrieving events from 1970 to 2017. Four regressors (linear models, gradient boosting machines, deep learning and random forests) and ensembles of them were been applied for predicting the maximum magnitude in the coming seven days. Due to the high computational resources required, the use of big data technologies and infrastructures was necessary. Spark distributed computing framework and H_2O library for cluster computing in R language were been used in this approach. Finally, Amazon cloud infrastructures were been also used.

The rest of the paper is structured as follows. Section 2 reports and discusses the relevant works related to big data and earthquake prediction. Section 3 details the proposed methodology to apply regression algorithms and using cloud-based big data infrastructure to predict earthquakes in California. Reported results along with illustrative comments can be found in Section 4. Finally, the conclusions drawn from this work are summarized in Section 5.

2. Related works

Earthquake generation is an extraordinarily stochastic process. Determining the time, the location and the magnitude of the next coming earthquake is an extremely difficult task. Moreover, considering that no direct measure of the accumulated stress and the strength of the material can be currently made. This search has made researchers to develop many different models [12].

These models can be classified in two groups [13]: the probabilistic methods (based on analysing the seismicity distribution) and the methods based on artificial intelligence (based on learning from the time series data).

Among all the initiatives done, the Regional Earthquake Likelihood Model (RELM) generated up to 18 different models. Based on that the probability of the occurrence of an earthquake follows a Poisson distribution, Petersen et al. [14] created a time-independent model. A 24-hours forecast method was proposed by [15]. This probability model uses foreshock/aftershock statistics. An intermediate to long time probabilistic forecast model was developed by [16]. Simple methods for determining the long-term average seismicity were created by [17]. The most fruitful author was Ward [18] who proposed five methods. The first one is based on smoothed seismicity. The second one considers GPS derived strain and the Kostrov's formula. The third one uses geological fault slip-data. The next one is an average of the previous methods and the last one uses an earthquake simulator. Two different models for a 24-hours forecast were proposed by [19]. These considered that an earthquake can be triggered by earlier shakes or can trigger later events. A five-year smoothed-seismicity model for M$\geq$5.0 for southern California was developed by [20]. Similarly, a smoothed-seismicity model was presented by [21] and it is based on small earthquakes for mapping large earthquakes. Ebel et al. [22] generated a 5-year forecast for M$\geq$5.0. Moreover, they also presented two one-day forecast methods for earthquakes larger than or equal to 4.0. Finally, Rhoades [23] introduced a method for long-range forecasting, based on preceding minor earthquakes for forecasting large events.

Despite some successful predictions such as the Xiuyen prediction [24] and that at the Sanriku area in Japan on November 2001 [25], failures are dominant. The most significant failure is probably the one at Parkfield [26], due to the great effort and financial resources employed.

Recently, some promising models based on data mining have been proposed. In [27] the authors used clustering techniques to model seismic temporal data. This research was based on the previous work by [28]. Later, the M5' algorithm was used for relating the b-value and the occurrence of large earthquakes [29]. In [30] artificial neural networks were used for predicting earthquakes in Chile (a survey on artificial neural networks application to earthquake prediction can be found in [31]). A model for earthquake prediction on the seismogenic areas of the Iberian Peninsula was presented in [32]. The authors in [33] determined the best set of seismicity indicators to predict earthquakes. This work analyzed four zones of Chile (the most seismic country in the world) and it is based on the previous works of [34] and [30]. In [35] the authors produced a method to test the validity of the seismicity indicators used, closing the existing gap between geologists and data mining experts.

Random forests algorithm (RF) has been extensively used in literature. In [36], RF was used to predict the remaining time before the next failure derived from earthquakes. In that

work, RF identifies two classes of signals and uses them to predict failure: shear stress and dynamic strain encompassing two failure events, and a zoom of dynamic strain when failure is in the distant future. The work proposed in [37] addressed the prediction of large earthquakes (higher than or equal to magnitude 5.5) in the Hindukush region of Pakistan. Authors used artificial neural networks (ANN), recurrent neural networks, RF and a linear combination of tree classifiers named LPBoost, which maximizes a margin between training instances of different classes (binary classes).

A large set of classifiers were put in comparison in [38] for earthquake prediction in Italy. Specifically, Logit Boost, Bagging, Naive Bayes (NB), Bayes Net, Logistic regression, SV-Cm (a deep learning algorithm based on a supervised contractive map), MLP-Bp, C4.5, RF, KNN and Linear Regression. Despite the vast number of algorithms, the classification accuracy was between 30% and 40% for earthquakes with magnitudes larger than 3.0. In [39], five classifiers (SVM, ANN, KNN, C4.5 and NB) were used to analyze the predictability of earthquake datasets previously grouped by clustering. Such work proposed two studies: the first one analyzes the ability of the different groups to train general prediction models, the second analyzes the diversity and representativeness covered by the samples of the clusters of each group.

Classification of large earthquakes via ensemble learning was addressed in [40] up to magnitudes larger than 7.0 for Chile datasets. Different unbalanced techniques were analyzed, including undersampling and oversampling as preprocessing, along with ensembles as boosting and bagging. Finally, in [41], generalized linear models (GLM) were used in combination with a model fitting based on the AIC measure for predicting the probability of near-fault earthquake ground motion pulses.

The study of the state of the art reveals that machine learning algorithms have been recently used. However, explored datasets included limited information due to space and computational limitations that since machines exhibit. Therefore, the exploration of big data analytics in this context is justified and it is expected to serve as seed for future research works.

3. Methodology

The methodology used to carry out the proposed regression study is described in this section. It is a methodology through which a large amount of earthquake events are retrieved, divided and used to train and test a set of machine learning algorithms in a comparative way. The whole procedure is sustained by a big data infrastructure placed in a public cloud.

In Section 3.1 the entire procedure is summarized. The following five sections (3.2 to 3.6) explain in detail every phase of the methodology. Finally, Section 3.7 describes the underlying IT infrastructure used, which is based on public cloud and big data technologies.

3.1. Overall methodology

The methodology carried out in this work is shown in a schematic way in Figure 1. First, a large catalog of earthquake events is retrieved from a public place. The purpose of selecting a large catalog was to prove the ability of the Big Data infrastructure to address large amount of events and process them.

The catalog acquired corresponds to the state of California from 1970 to 2017. A grid of latitudes and longitudes was established covering the state of California. Such grid produces a cell matrix where each cell has a size of 0.5×0.5 (latitude $\times$ longitude).

Every cell of the grid has a number of events ($L_{i,j}$) and a magnitude range ($min_{i,j}$ - $max_{i,j}$). In order to select a subset of cells to perform a comparative regression study, two thresholds, μ_1 and μ_2, were applied to filter cells. Specifically, cells with $L_{i,j} \geq \mu_1$ and $max_{i,j} \geq \mu_2$ were selected producing 27 cells used to feed the study.

A set of 16 seismic features are generated from each selected cell resulting on a set of 27 regression datasets. The target prediction is the maximum magnitude in the next seven days. Every regression

dataset, which is sorted ascending by time, is divided in two parts: the first 75% for training models and the rest (25%) for testing purposes. Thus, 27 pairs training/test were produced.

A set of four machine learning-based regressors were used to carry out the regression study: generalized linear models (GLM), gradient boosting machines (GBM), deep learning (DL) and random forests (RF). The best performance was achieved by RF and, for such reason, four ensembles based on the stacking technique were built using RF as base learner: RF-GLM, RF-GBM, RF-DL and ALL (RF-GLM-GBM-DL).

Every regressor is trained using each training dataset producing a regression model. Models were then applied to each testing dataset resulting on earthquake predictions. All predictions are compared with their corresponding actual values producing a set of deviations. Such deviations are evaluated resulting on absolute and relative errors. Those errors are shown and discussed in Section 4.

3.2. Data acquisition and preparation

Earthquake data was acquired from the FTP site of the ANSS Composite Earthquake Catalog (`ftp://www.ncedc.org/pub/catalogs/anss`), through the Northern California Earthquake Data Center (NCEDC) [42] (last accessed on Apr 15, 2017).

Table 1 shows the characteristics of the catalog of events downloaded from the FTP site. The catalog has a size of 917.7 MB of decompressed text files. The catalog contains a file for each month in CNSS format. A time period from Jan, 1970 to Apr, 2017 was considered, resulting on a set of 568 files containing earthquake events for each month in the considered period.

The variables used from the catalog entries were the latitude, the longitude and the magnitude of the events. The catalog was filtered according to a minimum magnitude $M_0 = 2.5$. Thus, only events with at least such magnitude will be considered from this point to the rest of the work. This filtering resulted in 63,960 events with magnitude greater than or equal to M_0.

Figure 2 shows the location of the considered events that occur more frequently. Events with the highest frequency are colored in red. It can be noticed that there are up to seven zones of high seismic activity. Three of them are particularly recurrent: the Joseph D. Grant County Park (near to the city of San Jose), the Sierra National Forest (near to the city of Bishop) and the San Bernardino National Forest (near to the city of San Bernardino).

According to the magnitude of the events, Figure 3 shows a map with the location of the earthquakes and a color scale from 2.5 (yellow) to 7.3 (red) degrees in the Richter scale. More significative areas in terms of high magnitude are the same highlighted in Figure 2 plus the area of the Humboldt Redwoods State Park, at the north of the state of California.

3.3. Catalog grid and filtering

From the catalog of earthquakes described in the previous subsection, a grid of cells defined by latitudes and longitudes was built. Table 2 shows the grid configuration used in this work. Specifically, cells of the grid are squared and they have a fixed size of 0.5 degrees (grid granularity).

The grid starts in the coordinates computed by the floor of the minimum latitude and longitude from the entire catalog of events. Specifically, the minimum latitude and longitude of catalog events are 32.55 and -124.37, respectively. Therefore, the grid starts at coordinates (32, -125), or in GPS coordinates (32 N, 125 W). From this point, cells are counted each 0.5 degrees in both latitude and latitude directions until the maximum point of the catalog (41.99, -114.56), forming the grid of study.

Note that the events were previously filtered by M_0 and the geographic distribution of events is not uniform. For such reason, void cells (without events) can appear and, therefore, they were removed from the grid resulting on a set of 177 cells.

In order to analyze the most significant earthquakes in the presented regression study, only cells which contain at least 500 events and one or more events with more than 5 degrees of magnitude were considered (these filtering specifications are summarized in Table 2). Finally, as result of the indicated filters, 27 cells are considered for the regression study. Those cells contain from 538 to 5575 events inside.

A map of locations for the 27 cells of study is provided in Figure 4. These locations are placed in the previously described highest seismicity areas in the state of California from 1970 to 2017. Specifically, these locations are indicated in detail in Table 3.

The dataset names of the selected cells are assigned according to their corresponding cell coordinates. For example, dataset named 2-19 contains the events in the cell (2,19) of the grid. Table 3 shows for each dataset both the latitude and the longitude ranges (minimum and maximum) and its centroid of points (specified in columns Lat.Cen and Lon.Cen).

The magnitude distribution of events in the considered datasets is summarized in Table 4. Columns size, Q1, median, mean, Q3 and max show the number of events, the first quartile of magnitudes, median, mean, third quartile and the maximum magnitude of each dataset, respectively. As it can be noticed, datasets 6-18 and 17-2 have the highest values of earthquake magnitudes.

3.4. Feature generation

For each selected cell from the grid a propositional dataset was built containing a set of features to be served as input to further regression models. Along with these features an outcome variable (continuous class) is included in datasets indicating the maximum magnitude of events in the next week. In this subsection, the seismic input features are described.

In Table 5 the set of seismic features are enumerated. The definitions of all used seismic indicators were been taken from two previous works [34, 30]. Specifically, the features x_1, x_2, x_3, x_4, x_5, x_6 and x_7 were firstly introduced in [30] and b, a, η, ΔM, T, μ, c, $dE^{1/2}$ and M_{mean} were proposed in [34]. To assess the features of a given event, the previous n events are calculated. In this work n was been set to 50 events, as suggested in [43] and successfully used in [33, 32, 30]. All the attributes were been normalized between 0 and 1.

The seismic indicator b corresponds to the Gutenberg-Richter law's b-value [44]. The authors in [34] used the least squares method for calculating the b-value. Due to the lack of robustness of this method when large infrequent earthquakes happen, [30] used the maximum likelihood method which is described in Equation 1:

$$b = \frac{\log e}{(1/n)\sum_{j=0}^{n-1} M_{i-j} - M_0} \tag{1}$$

The parameters involved in the Equation 1 are the number of events considered prior to e_i, n; the magnitude of the event e_{i-j}, M_{i-j}; and the cutoff magnitude of the seismic zone, M_0. The rest of the seismic features (x_1, x_2, x_3, x_4, x_5, x_6, x_7, a, η, ΔM, T, μ, c, $dE^{1/2}$ and M_{mean}) are defined as they were proposed in [34, 30].

3.5. Regression algorithms

In this section, the machine learning-based algorithms used for the regression in the presented study are described. Five different approaches were considered: generalized linear models, gradient boosting machine, deep learning, random forests and stacking ensembles.

Generalized Linear Models (GLM) provides a flexible generalization of the ordinary multiple linear regression with error distribution models other than a Gaussian distribution. GLM unifies various other statistical models, including Poisson, linear, logistic, and others when using L1 and L2 regularization. Due to the problem nature of earthquakes magnitude prediction, response variable is continuous and, therefore, a Gaussian distribution was used.

GLM belongs to the most commonly-used models for many types of data analysis use cases. Some problems, specially linear ones, can be addressed successfully using GLM, but others may not be as accurate if the variables are more complex. Namely, when the response variable has a non-linear distribution or the effect of the input variables is not linear, GLM can produce results less accurate than other non-linear models.

Gradient Boosting Machines (GBM) is a decision tree-based algorithm which is an ensemble method. GBM makes iteratively more than one decision tree combining their outputs. Boosting is the ensemble technique used in GBM to produce and select the different decision trees. Boosting-based techniques give more importance to the harder-to-learn training data, it tends to reduce bias in its predictions. GBM focuses its attention on the difficult instances in the training data, the ones that are hard to learn. That is

favorable, but it can also be risky. If there is one outlier that each tree keeps getting wrong it is going to get boosted and boosted achieving the maximum importance. If such outlier is a real unusual event then learning procedure is adequate, but if it is a measuring error it can distort the GBM accuracy.

GBM produces a prediction model in the form of an ensemble of weak prediction models, fitting consecutive trees where each solves for the net error of the prior trees. GBM builds the model in a stage-wise fashion and is generalized by allowing an arbitrary differentiable loss function. Results of new trees are applied partially to the entire solution. GBM often produces the best possible model but it is necessary to set proper stopping points to avoid overfitting, because GBM is sensitive to noise and extreme values.

Whereas underlying concept of linear models-based GLM is mathematics and decision trees-based GBM is logic, Deep Learning (DL) is a black box inspired by the human brain. DL models create high-level abstractions in data by using non-linear transformations in a layered-based iterative procedure. DL can address both supervised and unsupervised learning, which can use unlabeled data and it is widely used for pattern recognition in images or speech.

DL is based on a Deep Neural Network (DNN) which, in turn, is an Artificial Neural Network (ANN) with multiple hidden layers between input and output layers. DNNs can model complex non-linear relationships and generate compositional models where patterns in data are expressed as a layered composition of primitives. The extra layers enable composition of features from previous layers giving the potential of modeling complex data. DNNs are typically designed as feedforward networks, but other architectures like recurrent or convolutional neural networks were also applied.

DL implementation used in this work is based on a multi-layer feed-forward ANN that is trained with stochastic gradient descent using back-propagation. The network contains a large number of hidden layers consisting of neurons with tanh activation function. Despite its high computational cost, it scales well with big data, because each compute node in the cluster trains asynchronously a copy of the global model parameters on its local data with multi-threading and contributes periodically to the global model across the network.

Random Forests (RF) is an ensemble of decision trees based on the bagging technique using bootstrap aggregation. The idea is avoiding overfitting in complex data sets considering wide sets of trees and using of them to perform predictions on new data. RF combine multiple decision trees, each fit to a random sample of the original data. For classification the most frequent response is returned, for regression the mean of each tree response is used.

RF do not train every tree with all training data, instead of this, different random samples of rows and columns are given to each tree. RF is non-linear and it is robust for noisy data. It is able to reduce variance when predicting non-seen data with minimal increase in bias. Moreover, RF has the advantage of having few parametrization, mainly the number of trees used in the ensemble.

Apart of ensemble-based methods GBM and RF, the stacking technique of ensemble learning was considered in the present study. Both boosting and bagging of GBM and RF, respectively, are ensembles that take a collection of weak learners and forms a single strong learner. Stacking technique involves the training of a second-level metalearner to ensemble a group of base learners. The metalearner algorithm learns the optimal combination of the base learner fits. Unlike boosting and bagging, the goal in stacking is to ensemble strong and diverse sets of learners together.

Stacking builds the ensemble by training each of a set of B base algorithms on the training set. It then performs a k-fold cross-validation on each of these base algorithms and collect the cross-validated predicted values from each of the B algorithms. The M cross-validated predicted values from each of the B algorithms are combined to form a dataset called *level-one* with M instances and B predictors plus the original outcome variable. Then the metalearner is trained with such dataset. The ensemble model consists of the B base learning models and the metalearner model, which can then be used to produce predictions on test sets.

3.6. Validation and evaluation

In order to analyze and compare the performance of the regression algorithms on the different datasets, a hold-out scheme of validation was used. Specifically, each dataset was split in two parts: the first one includes the 75% of data and it is used to train the models, the second one includes the remaining 25% and it is used to test the models.

Since 27 datasets were splitted, there were 27 training sets and 27 test sets, each one of them for each dataset. Regressors were trained with each training set separately producing 27 models, one for each training set. Each model was tested also separately with its corresponding test set, producing its predictions.

Evaluation metrics computed for predictions made with different models are described as follows. Due to the regression nature of the problem of earthquakes magnitude prediction, metrics computed are absolute (MAE) and Relative Errors (RE). Equations 2 and 3 show the formulas of such types of error.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|\hat{y}_i - y_i| \tag{2}$$

$$RE = \frac{1}{n \times max(y_i)}\sum_{i=1}^{n}|\hat{y}_i - y_i| \tag{3}$$

In equations 2 and 3, n is the number of predicted test instances, y_i is the actual outcome value (the magnitude of the maximum event in the next week) and $\hat{y}$ is the predicted value for the outcome variable. Due to non-significant events were removed ($y_i \geq M_0, \forall i = 1..n$), some instances do not have any event in the next week. In such cases, the outcome variable is zero. For this reason, standard relative errors like RAE or MAPE cannot be used because they divide deviations between actual values. Therefore, the RE is divided by the maximum magnitude in the dataset, which is enough significant information about the proportion of the magnitude range that errors represents.

3.7. Big Data infrastructure

To perform the regression study, all phases of the methodology were implemented on a cloud-based Big Data infrastructure. The use of Big Data technologies was necessary due to the high number of earthquake events considered in the study. The infrastructure implemented in this work is drawn in Figure 5.

Amazon Web Services (AWS) were selected to provide a platform as a service in which all software components needed are deployed in the proposed methodology. The first step is the data ingestion, that consists in loading the whole catalog of earthquake events into the AWS cloud. Such procedure was carried out importing ANSS Composite catalogs from FTP to Amazon S3 service directly (without passing by any local filesystems) using the import tool of the S3 on-line control panel.

Catalog files were parsed using an eventual instance of Amazon EC2 to feed structured data to an Amazon Redshift database. Fields parsed were the timestamp, the longitude, the latitude and the magnitude of the earthquake events. An Amazon EMR 5.5.0 cluster was launched with Hadoop 2.7.3 and Apache Spark 2.1.0. Once the cluster is running, R 3.4.0 and the H_2O 3.10.4.6 library were installed using the bundle Sparkling Water distribution for Apache Spark.

An R script was developed to build the grid of cells from the event catalog tables in Amazon Redshift. Such grid was filtered according to the procedure described in Section 3.3 producing the set of 27 selected cells of study. Those cells are stored in Amazon Redshift in a new table. Next, seismic features are generated for each selected cell and resulting datasets were split in training and test. Later, they were stored in a next table of Amazon Redshift. Such process was run by other R script inside the Amazon EMR cluster.

Once training and test datasets were built, the machine learning-based regressors (GLM, GBM, DL, RF and Stacking ensembles) implemented in the H_2O library were executed to train models from training datasets and to predict the earthquakes of the test splits. Finally, absolute and relative errors were computed from an R script inside the Amazon EMR cluster. Error tables were stored both in Amazon S3 using LaTeX format and in Amazon RDS. Different error plots were produced and stored in a specific Amazon S3 output bucket for plots.

4. Results and discussion

Prediction results of the proposed regression study are shown and discussed in this section. In order to measure and compare the effectiveness achieved by the different machine learning-based algorithms on

the 27 datasets described in the methodology section, the evaluation metrics previously introduced in such section were computed.

Tables 6 and 7 show the absolute and the relative errors, respectively, produced by the different methods for each dataset of study. Lowest errors are shown in bold text for each dataset. RF achieved the best performance on average with a mean absolute error of 0.74 degrees in the Richter scale. Moreover, RF was the most effective for each dataset in the 81% of cases.

GLM and DL had the worst performance, producing up to 2.50 and 4.04 absolute deviations, respectively. To see their behaviors in detail, Figure 6 shows the dispersion of errors using a boxplot representation for each regressor. As it can be noticed, GLM and DL had the highest error dispersion.

GBM achieved competitive performance when predicting earthquakes magnitudes in regression, obtaining the highest accuracy in three of the datasets (3-18, 5-16 and 9-9). Moreover, its error dispersion is low and similar to that achieved by RF, as it can be seen in Figure 6.

All stacking ensemble combinations were carried out, but only those which includes RF were shown. All other ensembles (GLM-GBM, GLM-DL, GBM-DL and GLM-GBM-DL) performed worse and, to be concise, they were omitted from the study. All presented ensembles performed similarly, they achieved the same absolute error average (0.76) and showed very similar error dispersion.

In order to analyze in detail the sign of deviations produced by regressors in their predictions, Figure 7 shows histograms of errors for each regressor. All RF-based algorithms (RF, GLM-RF, GBM-RF, DL-RF, ALL) show the same behavior in the sign of their errors. They produced higher quantity of positive errors than negative ones (positive bias), as it can be noticed in the histograms of Figure 7. Precisely, stacking versions of RF produced higher positive errors than RF and lower negative errors.

RF-based stacking ensembles did not overcome to the base RF algorithm. This could be due to a lack of error complementation among RF and other methods. Specifically, instances in the cross-validated procedure of stacking formation predicted inaccurately by RF were not better predicted by any other algorithm in the ensemble. For such reason, stacking ensembles performed very similar to the base RF in the test sets.

GBM shown less bias in its error signs. They are more equilibrated around zero error and it approaches to a Gaussian distribution of its errors, which is desirable for a regressor. This behavior was expected due to nature of boosting (GBM) versus bagging (RF) ensembles. Boosting methods tend to decrease bias while bagging techniques tend to reduce variance increasing its bias, as it is mentioned in Section 3.5.

GLM and, specially, DL produced eventually very high errors (error outliers), up to -7.86 in GLM and -16.21 in DL. GLM could be too simple for this problem, because future earthquake magnitudes depend on very complex non-linear relationships of input seismic features.

The case of DL is different, it is the model with the highest complexity of the comparative. Its low accuracy could be due to its high parametrization. It has the highest number of parameters among the analyzed regressors. It could be improved performing a previous parameter tuning using training sets. In this work, the default configuration of deep learning implementation in the H_2O library was used.

A detailed analysis of predictions was performed according to the different magnitude values in order to see the performance of the different regressors on larger magnitudes, which are more complex to be accurately predicted. To carry out this study, all predictions performed for 25%-test splits of the 27 studied datasets were considered. Then, predictions were divided by regressor and classified by a set of intervals of the actual magnitude, ranging from the interval [0,3) to [7,8] (six intervals of size 1).

The complete set of predictions analyzed contains 90,920 samples and the maximum actual magnitude was 7.3. Tables 8 and 9 show mean absolute and mean squared errors, respectively; all averaged for each regressor and magnitude interval (mean and standard deviations were computed). The last two rows of these tables indicate the number (#) and percentage (%) of samples predicted, respectively, for each magnitude interval. Note that magnitudes lower than 5 suppose the 92.74% of the samples, leaving remaining events with larger magnitudes under-represented and, therefore, specially difficult to be predicted by machine learning techniques.

As it can be seen in Tables 8 and 9, the range of magnitudes [3,7) was reasonably well predicted with mean absolute errors lower than 0.6 with RF. Specially, range [4,7) was the most accurate in which errors were lower than or equal to 0.26. Extreme intervals [0,3)

and [7,8] were predicted with higher errors (up to 2.03 of MAE with RF in the case of the last interval). Mean squared errors (Table 9) reveal that ensemble methods were the best performance regressors for extreme intervals [0,4) and [7,8]. Such result could suggest possible improvement for large-magnitude predictions (magnitudes larger than 7) using more complex ensembles.

Figure 8 shows the scatter plots of actual-vs-predicted points for each regressor. A blue line was included in the scatter plots indicating the mean predicted value for each actual magnitude. As it can be seen, both GLM and DL perform worst due to their predicted values were significantly lower than the actual ones for large magnitudes. By contrast, GBM, RF and ensembles were sensible to higher magnitudes showing high correlation between actual and predicted values (up to $R^2 = 0.80$ with RF).

Table 10 shows execution times consumed for each process carried out in the proposed methodology. All regressors are parallelized in H_2O and they were executed in batch mode as Amazon EMR tasks. As expected, training process was the most consuming task (6 hours). DL and DL-RF were the slowest regressors in their training phase, consuming 2.4 hours together. The faster regressor was GBM consuming only 10 minutes. RF, the most effective regressor previously analyzed, was also one of the faster methods, taking only 18 minutes to train the models for all datasets.

5. Conclusions

Data from California earthquakes from 1970 to 2017 were been analyzed in this work. A total of 1 GB of information, divided into 27 datasets that identify cells of $0.5° \times 0.5°$, were been processed by means of cloud based infrastructure. In particular, distributed implementations from H_2O package of four popular regression methods were been used to predict earthquakes magnitude within the next seven days. As next step, stacking-based ensemble learning was been applied, reporting relative errors verging on 10% and absolute errors verging on 0.5. Methods based on trees performed better and these methods reached, in general terms, lower regression errors. In conclusion, the use of big data analytics in the field of earthquakes magnitude prediction opens a very promising research line that may help in simultaneously processing massive data with huge number of variables.

Acknowledgements

The authors would like to thank the Spanish Ministry of Economy and Competitiveness, Junta de Andalucía for the support under projects TIN2014-55894-C2-R and P12-TIC-1728, respectively.

References

[1] T. Aven. On how to define, understand and describe risk. *Reliability Engineering and System Safety*, 95(6):623–631, 2010.
[2] E. Florido, F. Martínez-Álvarez, A. Morales-Esteban, J. Reyes, and J. L. Aznarte. Detecting precursory patterns to enhance earthquake prediction in Chile. *Computers and Geosciences*, 76:112–120, 2015.
[3] A. Spicák and J. Vanek. Earthquake swarms reveal submarine magma unrest induced by distant mega-earthquakes: Andaman Sea region. *Journal of Asian Earth Sciences*, 116:155–163, 2016.
[4] C. Cecioni, G. Bellotti, A. Romano, A. Abdolali, P. Sammarco, and L. Franco. Tsunami Early Warning System based on Real-time Measurements of Hydro-acoustic Waves. *Procedia Engineering*, 70:311–320, 2014.
[5] D. K. Keefer. Lansdlides caused by earthquakes. *Bulletin of the Seismological Society of America*, 95(4):406–421, 1984.
[6] R. Verdugo and J. González. Liquefaction-induced ground damages during the 2010 Chile earthquake. *Soil Dynamics and Earthquake Engineering*, 79(B):280–295, 2015.
[7] L. F. Sá, A. Morales-Esteban, and P. Durand. A Seismic Risk Simulator for Iberia. *Bulletin of the Seismological Society of America*, 106(3):1198–1209, 2016.
[8] C. W. Tsai, C. F. Lai, H. C. Chao, and A. V. Vasilakos. Big data analytics: a survey. *Journal of Big Data*, 2:21:1–32, 2015.
[9] J. C. Jackson, V. Vijayakumar, A. Quadir, and C. Bharathi. Survey on Programming Models and Environments for Cluster, Cloud, and Grid Computing that Defends Big Data. *Procedia Computer Science*, 50:517–523, 2015.

[10] X. Romão, E. Paupério, and N. Pereira. A framework for the simplified risk analysis of cultural heritage assets. *Journal of Cultural Heritage*, 20:696–708, 2014.
[11] Q. Wang, D. D. Jackson, and Y. Y. Kagan. California earthquakes, 1800-2007: A unified catalog with moment magnitudes, uncertainties, and focal mechanisms. *Seismological Research Letters*, 80(3):446–457, 2009.
[12] K. F. Tiampo and R. Shcherbakov. Seismicity-based earthquake forecasting techniques: Ten years of progress. *Tectonophysics*, 522-523:89–121, 2012.
[13] G. Asencio-Cortés, F. Martínez-Álvarez, A. Morales-Esteban, and A. Troncoso. Medium-large earthquake magnitude prediction in Tokyo with artificial neural networks. *Neural Computing and Applications*, 28(5):1043Ű1055, 2017.
[14] M. D. Petersen, T. Cao, K. W. Campbell, and A. D. Frankel. Time-independent and time-dependent seismic hazard assessment for the state of California: Uniform California earthquake rupture forecast model 1.0. *Seismological Research Letters*, 78(1):99–109, 2007.
[15] M. C. Gerstenberguer, L. M. Jones, and S. Wiemer. Short-term aftershock probabilities: Case studies in California. *Seismological Research Letters*, 78(1):66–77, 2007.
[16] Z. Z. Shen, D. D. Jackson, and Y. Y. Kagan. Implications of geodetic strain rate for future earthquakes, with a five-year forecast of M5 earthquakes in southern California. *Seismological Research Letters*, 78(1):116–120, 2007.
[17] P. Bird and Z. Liu. Seismic hazard inferred from tectonics: California. *Seismological Research Letters*, 78(1):37–48, 2007.
[18] S. N. Ward. Methods for evaluating earthquake potential and likelihood in and around California. *Seismological Research Letters*, 78(1):121–133, 2007.
[19] R. Console, M. Murru, F. Catalli, and G. Falcone. Real time forecasts through an earthquake clustering model constrained by the rate-and-state constitutive law: comparison with a purely stochastic ETAS model. *Seismological Research Letters*, 78(1):49–56, 2007.
[20] Y. Y. Kagan, D. D. Jackson, and Y. Rong. A testable five-year forecast of moderate and large earthquakes in southern California based on smoothed seismicity. *Seismological Research Letters*, 78(1):94–98, 2007.
[21] A. Helmstetter, Y. Y. Kagan, and D. D. Jackson. High-resolution time-independent grid-based forecast for M=5 earthquakes in California. *Seismological Research Letters*, 78(1):78–86, 2007.
[22] J. E. Ebel, D. W. Chambers, A. L. Kafka, and J. A. Baglivo. Non-poissonian earthquake clustering and the hidden Markov model as bases for earthquake forecasting in California. *Seismollogical Research Letters*, 78(1):57–65, 2007.
[23] D. A. Rhoades. Application of the EEPAS model to forecasting earthquakes of moderate magnitude in southern California. *Seismological Research Letters*, 78(1):110–115, 2007.
[24] S. D. Zhang. *The 1999 Xiuyen-Haichung, Liaoning, M5.4 earthquake*. Beijing Seismological Press, 2004.
[25] T. Matsuzawa, T. Igarashi, and A. A. Hasegawa. Characteristic small-earthquake sequence off Sanriku, northeastern Honshu, Japan. *Geophysical Research Letters*, 29(11):1543–147, 2002.
[26] W. H. Bakun, B. Aagaard, B. Dost, W. L. Ellsworth, J. L. Hardebeck, R. A. Harris, C. Ji, M. J. S. Johnston, J. Langbein, J. J. Lienkaemper, A. J. Michael, J. R. Murray, R. M. Nadeau, P. A. Reasenberg, M. S. Reichle, E. A. Roeloffs, A. Shakal, R. W. Simpson, and F. Waldhauser. Implications for prediction and hazard assessment from the 2004 Parkfield earthquake. *Nature*, 437:969–974, 2005.
[27] A. Morales-Esteban, F. Martínez-Álvarez, A. Troncoso, J. L. de Justo, and C. Rubio-Escudero. Pattern recognition to forecast seismic time series. *Expert Systems with Applications*, 37(12):8333–8342, 2010.
[28] P. Nuannin, O. Kulhanek, and L. Persson. Spatial and temporal b value anomalies preceding the devastating off coast of nw sumatra earthquake of december 26, 2004. *Geophysical Research Letters*, 32, 2005.
[29] F. Martínez-Álvarez, A. Troncoso, A. Morales-Esteban, and J. C. Riquelme. Computational intelligence techniques for predicting earthquakes. *Lecture Notes in Artificial Intelligence*, 6679(2):287–294, 2011.
[30] J. Reyes, A. Morales-Esteban, and F. Martínez-Álvarez. Neural networks to predict earthquakes in Chile. *Applied Soft Computing*, 13(2):1314–1328, 2013.
[31] E. Florido, J. L. Aznarte, A. Morales-Esteban, and F. Martínez-Álvarez. Earthquake magnitude prediction based on artificial neural networks: A survey. *Croatian Operational Research Review*, 7(2):687–700, 2016.
[32] A. Morales-Esteban, F. Martínez-Álvarez, and J. Reyes. Earthquake prediction in seismogenic areas of the Iberian Peninsula based on computational intelligence. *Tectonophysics*, 593:121–134, 2013.
[33] F. Martínez-Álvarez, J. Reyes, A. Morales-Esteban, and C. Rubio-Escudero. Determining the best set of seismicity indicators to predict earthquakes. Two case studies: Chile and the Iberian Peninsula. *Knowledge-Based Systems*, 50:198–210, 2013.
[34] A. Panakkat and H. Adeli. Neural network models for earthquake magnitude prediction using multiple seismicity indicators. *International Journal of Neural Systems*, 17(1):13–33, 2007.
[35] G. Asencio-Cortés, F. Martínez-Álvarez, A. Morales-Esteban, and J. Reyes. A sensitivity study of seismicity indicators in supervised learning to improve earthquake prediction. *Knowledge-Based Systems*, 101:15–30, 2016.
[36] Bertrand Rouet-Leduc, Claudia Hulbert, Nicholas Lubbers, Kipton Barros, Colin Humphreys, and Paul A Johnson. Machine learning predicts laboratory earthquakes. *arXiv preprint arXiv:1702.05774*, 2017.
[37] KM Asim, F Martínez-Álvarez, A Basit, and T Iqbal. Earthquake magnitude prediction in hindukush region using machine learning techniques. *Natural Hazards*, 85(1):471–486, 2017.
[38] P. M. Buscema, G. Massini, and G. Maurelli. Artificial Adaptive Systems to predict the magnitude of earthquakes. *Bollettino di Geofisica Teorica ed Applicata*, 56(2):227–256, 2015.
[39] Gualberto Asencio-Cortés, Sanja Scitovski, Rudolf Scitovski, and Francisco Martínez-Álvarez. Temporal analysis of croatian seismogenic zones to improve earthquake magnitude prediction. *Earth Science Informatics*, pages 1–18, 2017.
[40] Manuel Jesús Fernández-Gómez, Gualberto Asencio-Cortés, Alicia Troncoso, and Francisco Martínez-Álvarez. Large earthquake magnitude prediction in chile with imbalanced classifiers and ensemble learning. *Applied Sciences*, 7(6):625,

2017.

[41] Shrey K Shahi and Jack W Baker. Regression models for predicting the probability of near-fault earthquake ground motion pulses, and their period. *Applications of Statistics and Probability in Civil Engineering*, 30(4):459, 2011.

[42] UC Berkeley Seismological Laboratory, United States Geological Survey (USGS), Calpine and Unocal Corporations. Northern california earthquake data center, 2014.

[43] P. Nuannin. The potential of b-value variations as earthquake precursors for small and large events. Technical Report 183, Uppsala University, Sweden, 2006.

[44] B. Gutenberg and C. F. Richter. Frequency of earthquakes in California. *Bulletin of the Seismological Society of America*, 34:185–188, 1944.

Tables

Zone	State of California
Period	1970-2017
Source	ANSS Composite Earthquake Catalog
Size	917.7 MB
Events	1,421,691 events
Max. magnitude	7.3

Table 1: Specifications of the original catalog used in this work from whose the different datasets for the regression study were extracted.

Lat/Long. granularity	0.5 degrees
Max. magn. threshold	5
Min. events/cell	500 events
Num. selected cells	27 cells
Resulting size range	538 - 5,575 events

Table 2: Definition of the grid established over the original catalog of California. Each selected cell from the grid produces a dataset to perform earthquake predictions in the presented regression study.

Dataset	Lat.Min	Lat.Cen	Lat.Max	Lon.Min	Lon.Cen	Lon.Max
2-19	32.50	32.83	33.00	-116.00	-115.73	-115.50
3-18	33.00	33.29	33.50	-116.50	-116.29	-116.00
3-19	33.00	33.13	33.50	-116.00	-115.68	-115.50
4-17	33.50	33.81	34.00	-117.00	-116.73	-116.50
4-18	33.50	33.89	34.00	-116.50	-116.28	-116.00
5-13	34.00	34.32	34.50	-119.00	-118.62	-118.50
5-14	34.00	34.30	34.50	-118.50	-118.38	-118.00
5-16	34.00	34.13	34.50	-117.50	-117.25	-117.00
5-17	34.00	34.22	34.50	-117.00	-116.78	-116.50
5-18	34.00	34.22	34.50	-116.50	-116.38	-116.00
6-17	34.50	34.74	35.00	-117.00	-116.72	-116.50
6-18	34.50	34.68	35.00	-116.50	-116.32	-116.00
8-8	35.50	35.70	36.00	-121.50	-121.10	-121.00
8-9	35.50	35.68	36.00	-121.00	-120.80	-120.50
8-15	35.50	35.80	36.00	-118.00	-117.70	-117.50
9-9	36.00	36.23	36.50	-121.00	-120.76	-120.50
9-10	36.00	36.21	36.50	-120.50	-120.28	-120.00
9-15	36.00	36.12	36.50	-118.00	-117.79	-117.50
10-7	36.50	36.88	37.00	-122.00	-121.60	-121.50
10-8	36.50	36.67	37.00	-121.50	-121.24	-121.00
11-7	37.00	37.25	37.50	-122.00	-121.72	-121.50
11-13	37.00	37.45	37.50	-119.00	-118.73	-118.50
12-7	37.50	37.75	38.00	-122.00	-121.85	-121.50
12-13	37.50	37.61	38.00	-119.00	-118.85	-118.50
12-14	37.50	37.56	38.00	-118.50	-118.44	-118.00
14-5	38.50	38.80	39.00	-123.00	-122.79	-122.50
17-2	40.00	40.36	40.50	-124.50	-124.23	-124.00

Table 3: Location of the 27 datasets analyzed in the study. Each dataset was named using the number of the cell in a grid drawn over the state of California. The latitude and longitude ranges (minimum and maximum) are shown along with the mean of latitudes and longitudes (centroid) of the events inside each dataset (Lat.Cen, Lon.Cen).

Dataset	Size	Q1	Median	Mean	Q3	Max
2-19	2195	2.63	2.81	2.95	3.15	5.80
3-18	1051	2.62	2.79	2.91	3.09	5.43
3-19	1950	2.62	2.81	2.94	3.12	6.60
4-17	1065	2.59	2.73	2.84	2.97	6.00
4-18	1386	2.60	2.77	2.90	3.07	6.10
5-13	1022	2.62	2.80	2.97	3.15	6.70
5-14	1281	2.68	2.99	3.11	3.40	6.60
5-16	889	2.59	2.74	2.87	3.01	5.60
5-17	2326	2.61	2.78	2.92	3.08	6.30
5-18	3013	2.60	2.76	2.90	3.01	7.30
6-17	1606	2.60	2.75	2.85	2.98	5.26
6-18	1827	2.62	2.80	2.93	3.10	7.10
8-8	763	2.64	2.85	2.98	3.23	6.50
8-9	717	2.63	2.80	2.92	3.10	5.00
8-15	1281	2.61	2.77	2.89	3.02	5.75
9-9	1346	2.65	2.85	2.96	3.14	5.40
9-10	1840	2.65	2.87	3.00	3.21	6.70
9-15	1366	2.61	2.77	2.90	3.04	5.30
10-7	1940	2.69	2.92	3.02	3.22	5.40
10-8	5575	2.66	2.89	3.00	3.20	5.50
11-7	1615	2.63	2.82	2.95	3.11	6.90
11-13	1595	2.68	2.93	3.00	3.20	6.10
12-7	807	2.62	2.80	2.91	3.06	5.80
12-13	4002	2.68	2.95	3.04	3.26	6.20
12-14	724	2.65	2.89	3.04	3.28	6.40
14-5	3156	2.60	2.75	2.86	3.00	5.01
17-2	538	2.64	2.84	2.94	3.10	7.20

Table 4: Magnitude distributions and sizes of the 27 datasets analyzed in the study. The size is shown as the number of events included in each dataset. The magnitude distribution is expressed as its first quartile, median, mean, third quartile and the maximum magnitude for each dataset.

Feature	Description
b	Gutenberg-Richter law's b-value
x_1	Increment of b between the events i and $i-4$
x_2	Increment of b between the events $i-4$ and $i-8$
x_3	Increment of b between the events $i-8$ and $i-12$
x_4	Increment of b between the events $i-12$ and $i-16$
x_5	Increment of b between the events $i-16$ and $i-20$
x_6	Maximum magnitude from the events recorded during the last week (OU's law)
x_7	Probability of recording an event with magnitude larger or equal to 6.0 using a probability density function
a	Gutenberg-Richter law's a-value
η	Mean square deviation
ΔM	Magnitude deficit
T	Elapsed time
μ	Mean time
c	Coefficient of variation
$dE^{1/2}$	Rate of square root of seismic energy
M_{mean}	Mean magnitude

Table 5: Set of features computed from catalogs to produce the datasets used to train and test the regression methods of the study.

Dataset	GLM	GBM	DL	RF	GLM-RF	GBM-RF	DL-RF	ALL
2-19	0.93	0.62	0.79	**0.56**	**0.56**	**0.56**	**0.56**	**0.56**
3-18	1.38	**1.33**	1.34	1.34	1.40	1.39	1.39	1.38
3-19	1.10	0.87	1.36	0.81	**0.80**	0.82	**0.80**	0.82
4-17	1.33	1.09	1.23	**1.05**	1.08	1.08	1.08	1.08
4-18	0.81	0.60	0.84	**0.59**	**0.59**	**0.59**	**0.59**	**0.59**
5-13	0.62	0.53	0.78	**0.46**	0.48	0.47	**0.46**	0.48
5-14	2.50	0.59	4.04	**0.57**	0.58	**0.57**	0.88	0.84
5-16	1.38	**1.30**	1.44	1.31	1.36	1.36	1.36	1.36
5-17	1.27	0.59	2.25	**0.55**	**0.55**	**0.55**	**0.55**	0.56
5-18	0.66	0.43	0.74	**0.37**	**0.37**	0.39	**0.37**	0.39
6-17	0.62	0.44	0.70	**0.41**	0.43	0.42	0.42	0.43
6-18	0.58	0.45	0.64	**0.40**	0.41	0.41	0.41	0.41
8-8	0.74	0.50	0.67	**0.46**	0.49	0.48	**0.46**	0.48
8-9	1.29	0.56	0.90	**0.53**	0.55	0.56	0.55	0.54
8-15	0.83	0.69	1.07	**0.65**	0.67	0.68	0.67	0.68
9-9	1.06	**0.87**	1.21	0.89	0.91	0.90	0.92	0.90
9-10	0.81	**0.59**	0.77	**0.59**	0.61	**0.59**	0.61	**0.59**
9-15	1.07	0.91	1.14	**0.89**	0.91	0.90	0.91	0.90
10-7	1.12	1.04	1.08	**1.02**	1.04	1.03	1.03	1.03
10-8	0.76	0.69	0.78	**0.64**	**0.64**	0.65	0.65	0.65
11-7	1.14	1.11	1.17	**1.08**	1.09	1.12	1.09	1.10
11-13	1.05	0.87	1.25	**0.84**	0.85	0.85	0.85	0.85
12-7	1.16	1.06	1.21	**1.05**	1.06	1.06	1.06	1.06
12-13	0.81	0.68	0.79	**0.61**	**0.61**	0.62	**0.61**	0.62
12-14	0.55	0.33	0.58	**0.30**	**0.30**	0.31	**0.30**	0.31
14-5	1.10	**1.09**	1.11	1.10	**1.09**	**1.09**	**1.09**	**1.09**
17-2	1.12	0.99	1.09	**0.95**	1.00	1.02	**0.95**	0.96
Average	1.03	0.77	1.15	**0.74**	0.76	0.76	0.76	0.76

Table 6: Mean absolute error of the different regressors analyzed when they predict the 27 earthquake datasets of the study. A hold-out validation scheme was applied splitting the first 75% of samples for training and the last 25% for testing. Regressors are generalized linear models (GLM), gradient boosting machines (GBM), deep learning (DL), random forests (RF) and 4 stacked ensembles including RF: GLM-RF, GBM-RF, DL-RF and GLM-GBM-DL-RF (ALL).

Dataset	GLM	GBM	DL	RF	GLM-RF	GBM-RF	DL-RF	ALL
2-19	0.16	0.11	0.14	**0.10**	**0.10**	**0.10**	**0.10**	**0.10**
3-18	0.25	**0.24**	0.25	0.25	0.26	0.26	0.26	0.25
3-19	0.17	0.13	0.21	**0.12**	**0.12**	**0.12**	**0.12**	**0.12**
4-17	0.22	**0.18**	0.21	**0.18**	**0.18**	**0.18**	**0.18**	**0.18**
4-18	0.13	**0.10**	0.14	**0.10**	**0.10**	**0.10**	**0.10**	**0.10**
5-13	0.09	0.08	0.12	**0.07**	**0.07**	**0.07**	**0.07**	**0.07**
5-14	0.38	**0.09**	0.61	**0.09**	**0.09**	**0.09**	0.13	0.13
5-16	0.25	**0.23**	0.26	**0.23**	0.24	0.24	0.24	0.24
5-17	0.20	**0.09**	0.36	**0.09**	**0.09**	**0.09**	**0.09**	**0.09**
5-18	0.09	0.06	0.10	**0.05**	**0.05**	**0.05**	**0.05**	**0.05**
6-17	0.12	**0.08**	0.13	**0.08**	**0.08**	**0.08**	**0.08**	**0.08**
6-18	0.08	**0.06**	0.09	**0.06**	**0.06**	**0.06**	**0.06**	**0.06**
8-8	0.11	0.08	0.10	**0.07**	**0.07**	**0.07**	**0.07**	**0.07**
8-9	0.26	**0.11**	0.18	**0.11**	**0.11**	**0.11**	**0.11**	**0.11**
8-15	0.14	0.12	0.19	**0.11**	0.12	0.12	0.12	0.12
9-9	0.20	**0.16**	0.22	**0.16**	0.17	0.17	0.17	0.17
9-10	0.12	**0.09**	0.12	**0.09**	**0.09**	**0.09**	**0.09**	**0.09**
9-15	0.20	**0.17**	0.21	**0.17**	**0.17**	**0.17**	**0.17**	**0.17**
10-7	0.21	**0.19**	0.20	**0.19**	**0.19**	**0.19**	**0.19**	**0.19**
10-8	0.14	0.13	0.14	**0.12**	**0.12**	**0.12**	**0.12**	**0.12**
11-7	**0.16**	**0.16**	0.17	**0.16**	**0.16**	**0.16**	**0.16**	**0.16**
11-13	0.17	**0.14**	0.21	**0.14**	**0.14**	**0.14**	**0.14**	**0.14**
12-7	0.20	**0.18**	0.21	**0.18**	**0.18**	**0.18**	**0.18**	**0.18**
12-13	0.13	0.11	0.13	**0.10**	**0.10**	**0.10**	**0.10**	**0.10**
12-14	0.09	**0.05**	0.09	**0.05**	**0.05**	**0.05**	**0.05**	**0.05**
14-5	**0.22**	**0.22**	**0.22**	**0.22**	**0.22**	**0.22**	**0.22**	**0.22**
17-2	0.16	0.14	0.15	**0.13**	0.14	0.14	**0.13**	**0.13**
Average	0.17	0.13	0.19	**0.12**	0.13	0.13	0.13	0.13

Table 7: Relative error of the different regressors analyzed when they predict the 27 earthquake datasets of the study. The relative error is computed as the mean absolute error divided by the maximum magnitude of each dataset. A hold-out validation scheme was applied splitting the first 75% of samples for training and the last 25% for testing. Regressors are generalized linear models (GLM), gradient boosting machines (GBM), deep learning (DL), random forests (RF) and 4 stacked ensembles including RF: GLM-RF, GBM-RF, DL-RF and GLM-GBM-DL-RF (ALL).

Regressor	[0,3)	[3,4)	[4,5)	[5,6)	[6,7)	[7,8]
GLM	1.25 ± 0.88	0.67 ± 0.64	0.57 ± 0.58	0.77 ± 0.81	1.15 ± 0.92	3.85 ± 0.49
GBM	1.20 ± 0.81	0.62 ± 0.67	0.30 ± 0.53	0.32 ± 0.62	0.34 ± 0.76	2.57 ± 1.99
DL	1.24 ± 0.87	0.79 ± 0.72	0.60 ± 0.60	0.80 ± 0.87	1.36 ± 1.00	3.49 ± 0.39
RF	**1.20±0.77**	**0.58±0.66**	**0.22±0.53**	**0.24±0.61**	**0.26±0.75**	**2.03±1.81**
GLM-RF	1.20 ± 0.78	0.58 ± 0.68	0.25 ± 0.53	0.26 ± 0.60	0.29 ± 0.74	2.07 ± 1.85
GBM-RF	1.20 ± 0.79	0.59 ± 0.66	0.25 ± 0.52	0.27 ± 0.60	0.28 ± 0.75	2.19 ± 1.83
DL-RF	1.20 ± 0.78	0.59 ± 0.67	0.24 ± 0.53	0.26 ± 0.60	0.28 ± 0.74	2.03 ± 1.83
ALL	1.20 ± 0.78	0.59 ± 0.66	0.25 ± 0.52	0.27 ± 0.60	0.28 ± 0.75	2.20 ± 1.83
# samples	35,608	27,168	21,544	5,936	616	48
% samples	39.16	29.88	23.7	6.53	0.68	0.05

Table 8: Mean absolute errors (mean ± std. deviation) produced for each regressor according to the different magnitude intervals.

Regressor	[0,3)	[3,4)	[4,5)	[5,6)	[6,7)	[7,8]
GLM	2.34 ± 2.79	0.85 ± 1.62	0.66 ± 1.63	1.25 ± 3.15	2.16 ± 3.87	15.04 ± 3.74
GBM	2.08 ± 2.40	0.83 ± 1.61	0.37 ± 1.44	0.49 ± 2.06	0.69 ± 2.93	9.91 ± 10.22
DL	2.28 ± 2.71	1.11 ± 1.89	0.72 ± 1.64	1.40 ± 3.33	2.84 ± 4.09	12.28 ± 2.67
RF	2.07 ± 2.29	0.80 ± 1.60	**0.33±1.42**	**0.42±2.15**	**0.62±2.66**	6.97 ± 7.72
GLM-RF	2.05 ± 2.25	**0.78±1.53**	0.34 ± 1.45	0.43 ± 2.11	0.62 ± 2.68	7.15 ± 7.85
GBM-RF	2.04 ± 2.23	**0.78±1.53**	0.34 ± 1.41	0.43 ± 2.05	0.63 ± 2.73	7.58 ± 8.13
DL-RF	2.05 ± 2.25	0.79 ± 1.55	0.33 ± 1.44	0.43 ± 2.11	0.62 ± 2.68	**6.88±7.57**
ALL	**2.03±2.25**	0.79 ± 1.53	0.34 ± 1.42	0.43 ± 2.06	0.63 ± 2.73	7.61 ± 8.14
# samples	35,608	27,168	21,544	5,936	616	48
% samples	39.16	29.88	23.7	6.53	0.68	0.05

Table 9: Mean squared errors (mean ± std. deviation) produced for each regressor according to the different magnitude intervals.

Process	Execution time
Data acquisition	15 minutes
Cell building	4 minutes
Cell selection	3 minutes
Feature generation	6 minutes
Dataset splitting	10 seconds
Training of the regressors	6 hours
Prediction of the test subsets	2 minutes
Statistics computation	1 minute

Table 10: Execution times of each process of the proposed methodology. These values are obtained with the IT infrastructure described in Section 3.7.

Figures

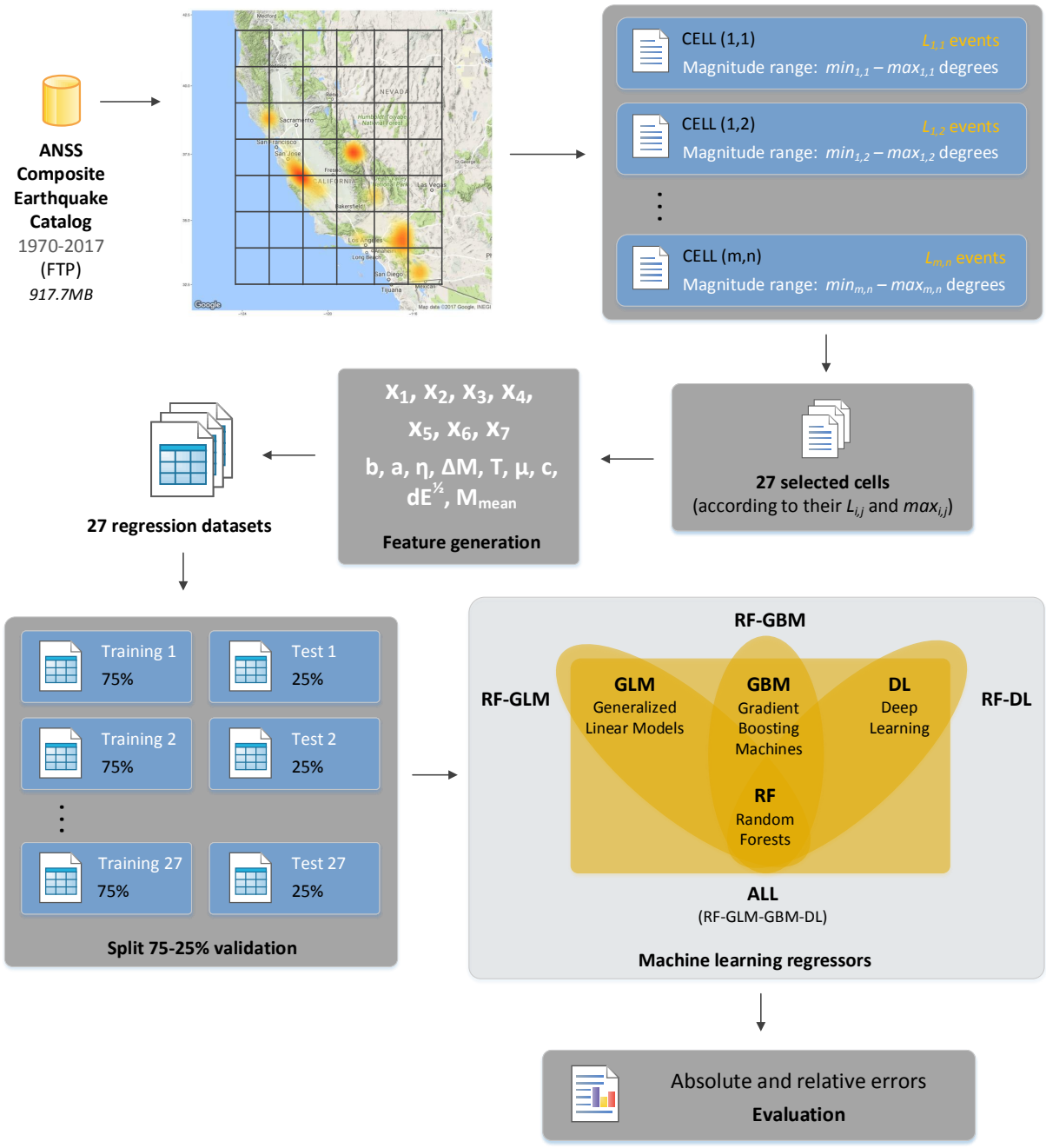

Figure 1: Schematic diagram of the methodology used to retrieve, divide, train and test the set of machine learning methods used to perform the proposed regression study.

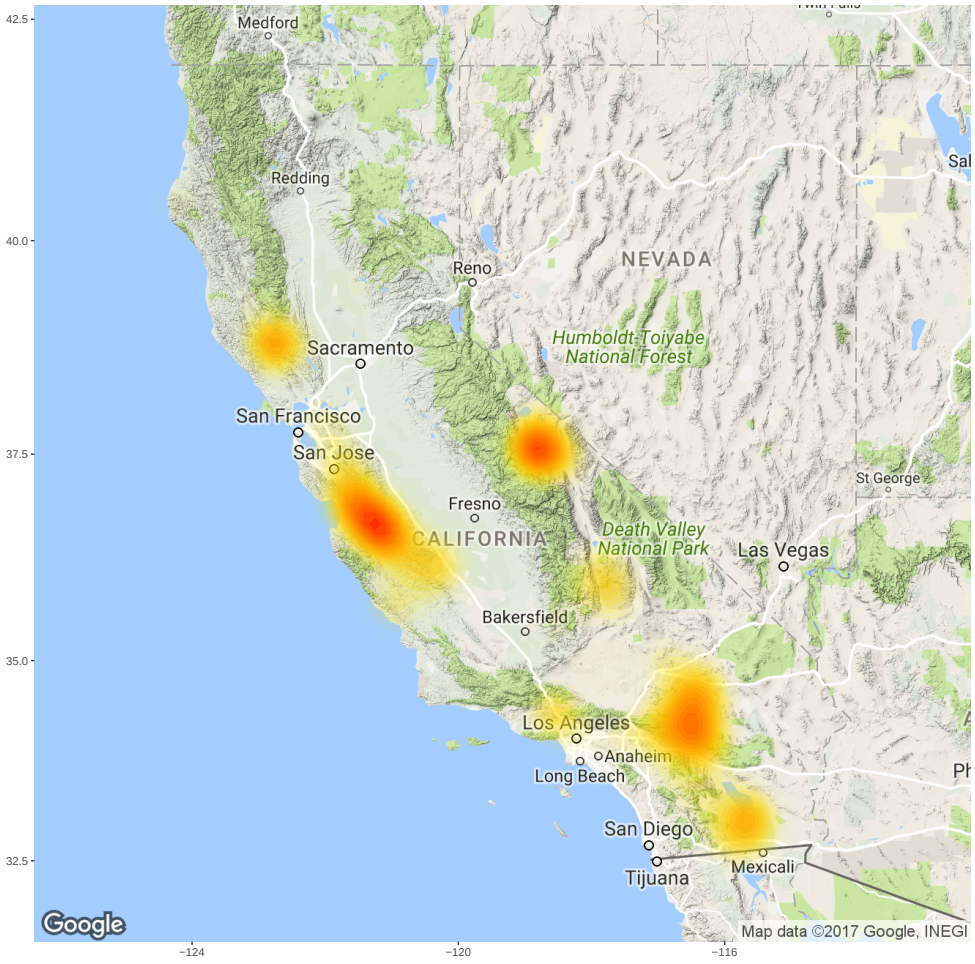

Figure 2: Earthquake events studied in California between 1970-2017 colored by the number of ocurrences. Data were obtained from the ANSS Composite Earthquake Catalog.

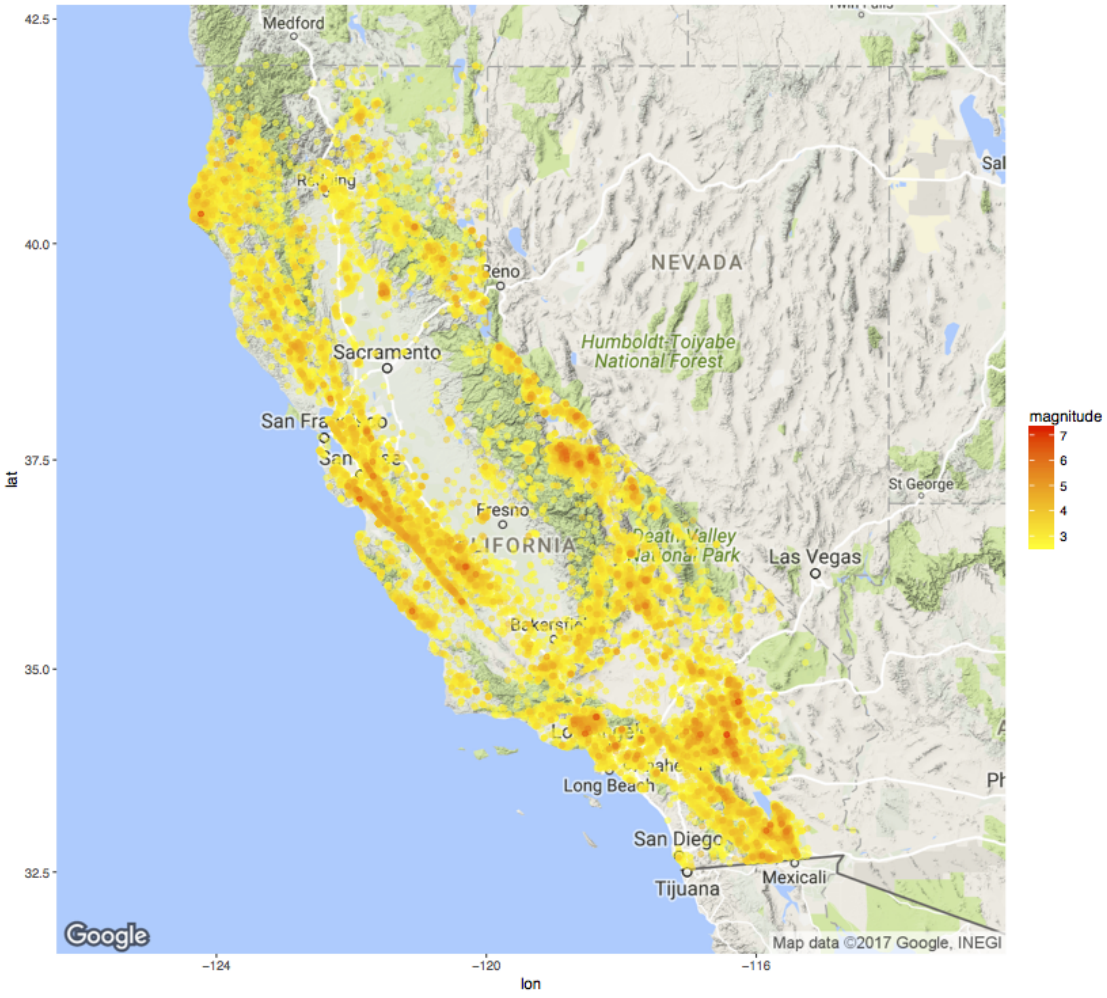

Figure 3: Earthquake events studied in California between 1970-2017 colored by their magnitude. Data were obtained from the ANSS Composite Earthquake Catalog.

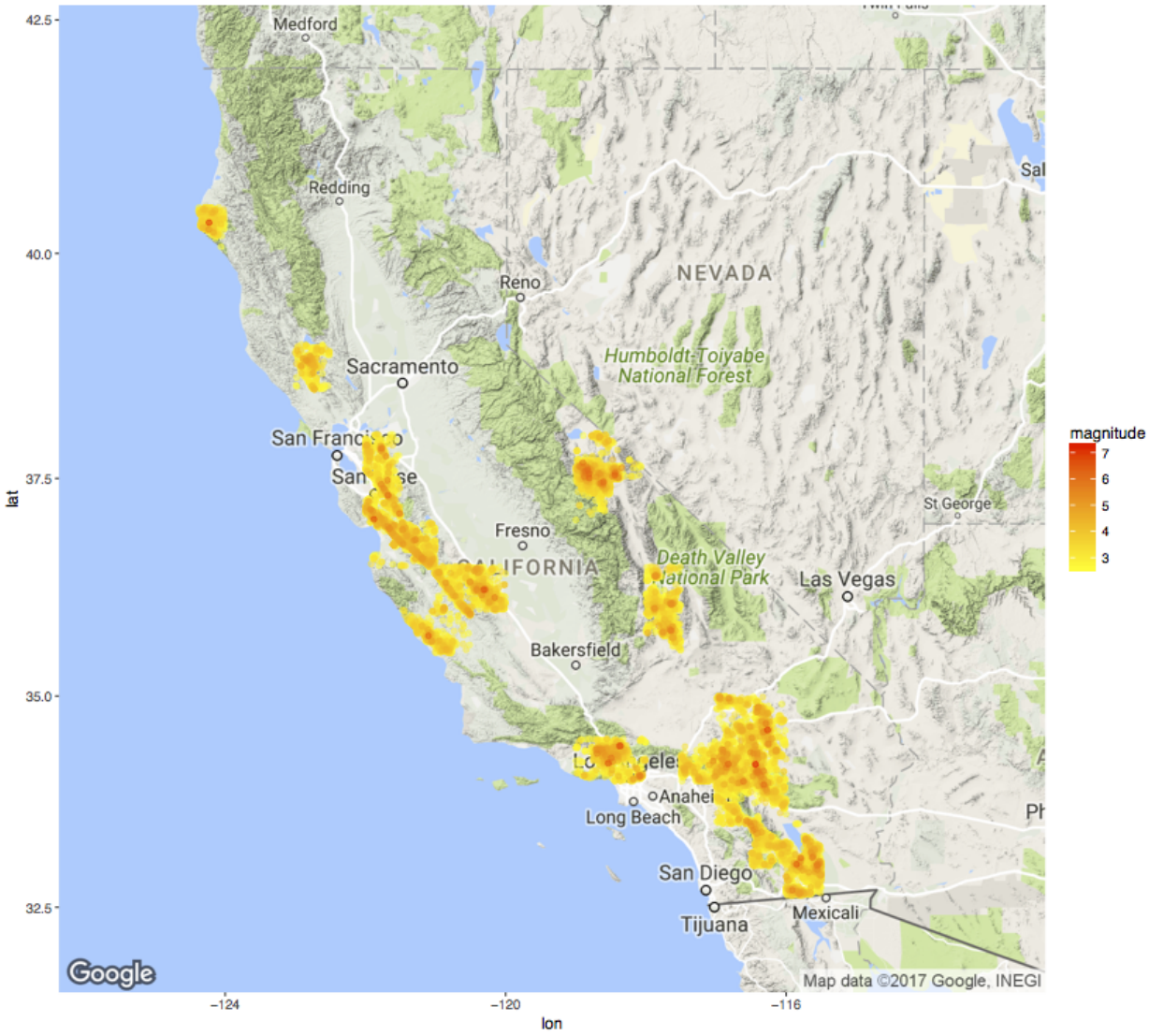

Figure 4: Earthquake events of the 27 filtered cells colored by their magnitude.

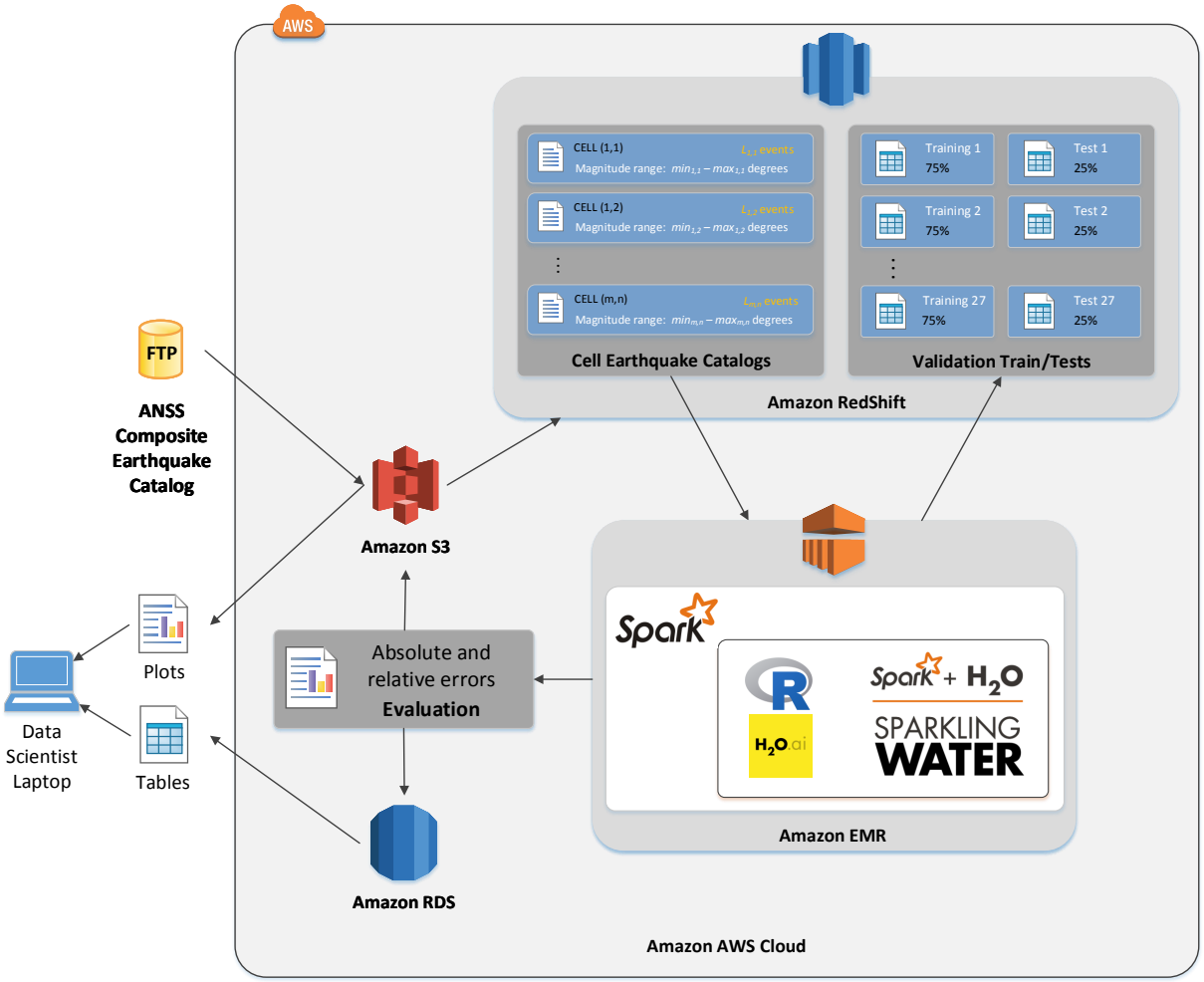

Figure 5: The cloud-based Big Data IT infrastructure implemented for the earthquake analysis and prediction study. Amazon Web Services were chosen to provide the platform as a service needed to deploy all software components of the proposed methodology.

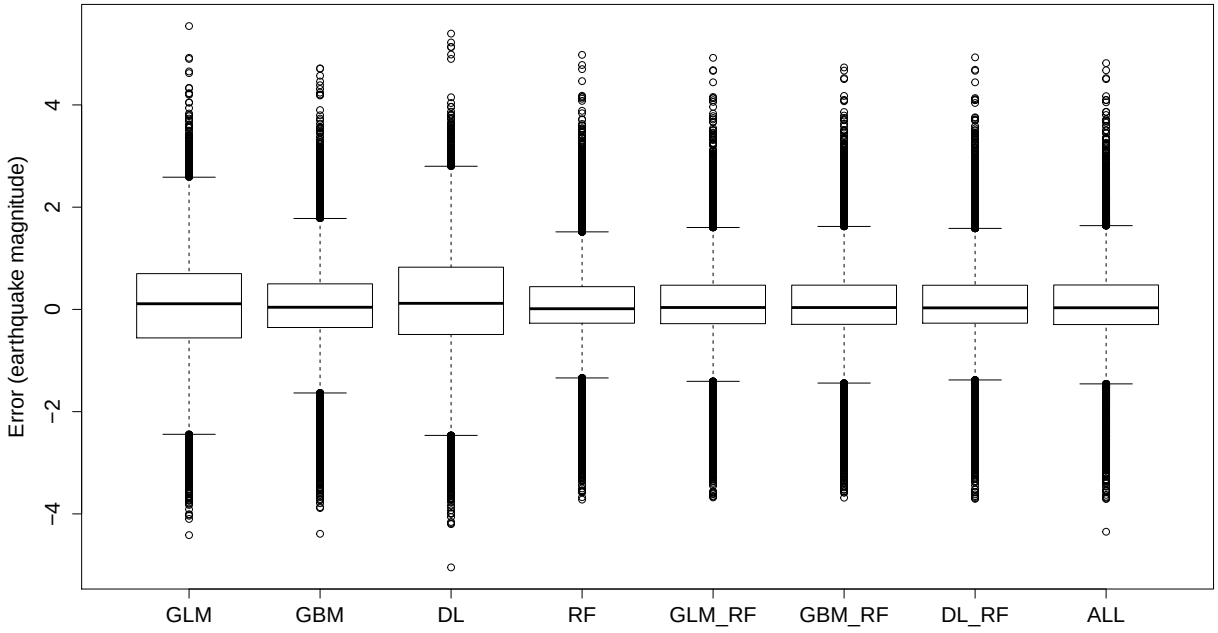

Figure 6: Boxplot of the errors produced by the regression algorithms when predicting the 27 datasets of the study.

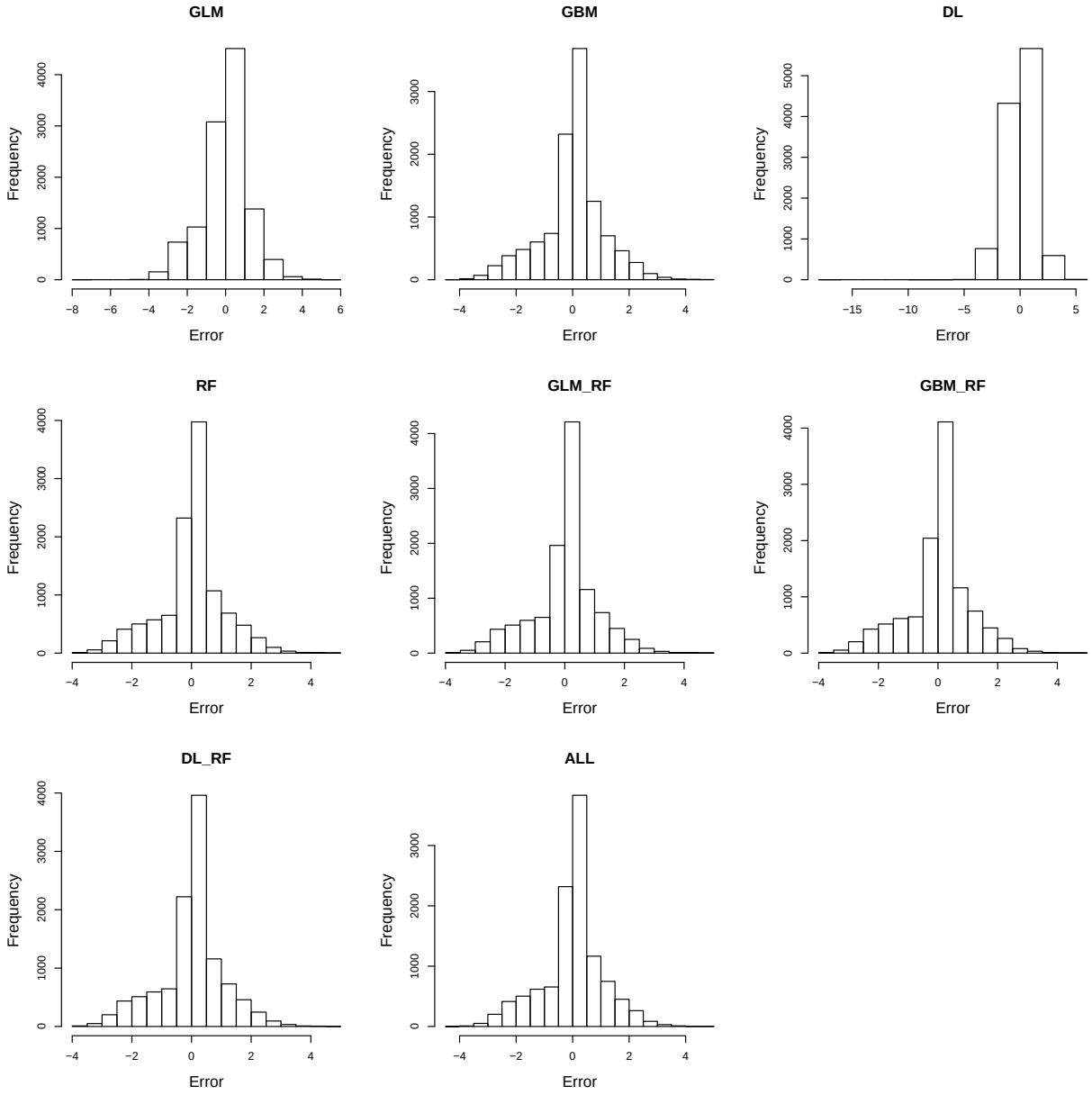

Figure 7: Histograms of the errors produced by the regression algorithms when predicting the 27 datasets of the study.

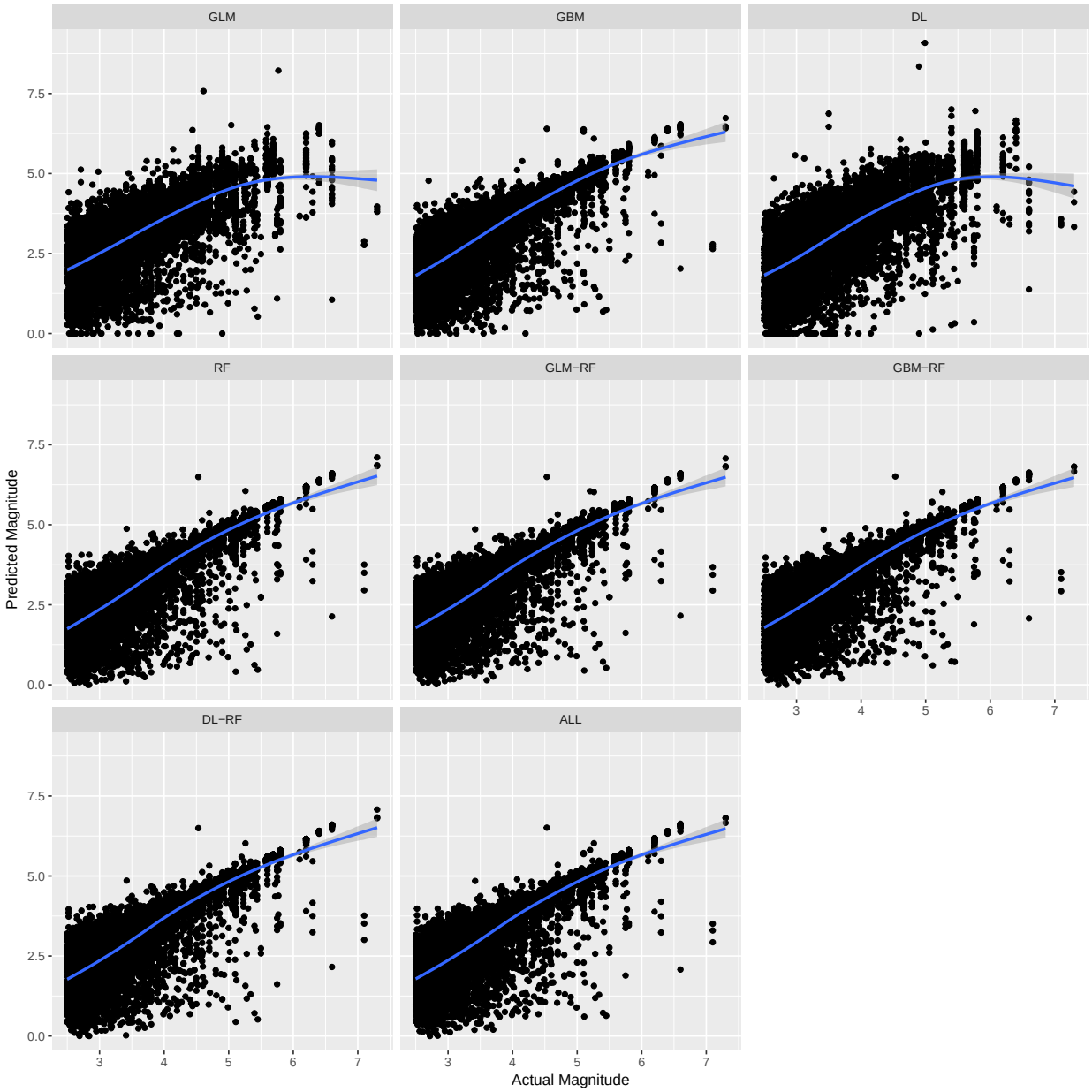

Figure 8: Actual versus predicted magnitude values for each regression algorithm when predicting the 27 datasets of the study.