



# Universidad de Montemorelos

Facultad de Ingeniería y Tecnología  
Ingeniería en Sistemas Computacionales

## Automated machine learning pipeline for geochemical analysis

Oscar A. Esteban Suarez

**1170541**

Asesor principal: Dr. Germán Harvey Alférez Salinas

Asesores secundarios: Dr. Benjamin L. Clausen y Dra. Ana Martínez Ardila

Montemorelos, Nuevo León, México

**5 de mayo de 2021**

## **Abstract**

Biplot diagrams are traditionally used for rock discrimination by means of utilizing geochemical data from samples. However, this approach has limitations when facing a high number of variables. Machine learning has been proposed as an alternative to analyze multivariate data for more than 70 years. However, the application of machine learning by geoscientists is still complicated since there are no tools that propose a pipeline that can be followed from preparing the data to evaluating the models. Automated machine learning aims to face this issue by automating the creation and evaluation of machine learning models. The contribution of this work is twofold. First, we propose a methodology that follows a pipeline for the application of supervised and unsupervised learning to geochemical data. Both methods were applied to a dataset of granitic rock samples from 6 blocks in the Peninsular Ranges and the Transverse Ranges Provinces in Southern California. For supervised learning, the Decision Trees model offered the best values to classify the samples from this region: accuracy: 87%; precision: 89%; recall: 89%; and F-score: 81%. For unsupervised learning, 2 components were related to pressure effects, and another 2 could be related to water effects. As a second contribution, we propose a web application that follows the proposed methodology to analyze geochemical data using automated machine learning. It allows data preparation using techniques such as imputation and upsampling, the application of supervised and unsupervised learning, and the evaluation of the models. All this without the need to program.

## **Index Terms**

Automated Machine Learning, Geochemistry, Machine Learning, Methodology, Southern California, Supervised Learning, Unsupervised Learning, Web Application

## CONTENTS

|                   |   |    |
|-------------------|---|----|
| <b>I</b>          | <b>Introduction</b>                     | 1  |
| I-A               | Background . . . . .                    | 1  |
| I-B               | Problem statement . . . . .             | 1  |
| I-C               | Justification . . . . .                 | 1  |
| I-D               | Objectives . . . . .                    | 1  |
| I-E               | Hypothesis . . . . .                    | 2  |
| <b>II</b>         | <b>Theoretical foundation</b>           | 2  |
| II-A              | Underpinnings of our approach . . . . . | 2  |
| II-B              | Related Work . . . . .                  | 3  |
| II-B1             | Supervised learning . . . . .           | 3  |
| II-B2             | Unsupervised learning . . . . .         | 3  |
| II-B3             | Discussion . . . . .                    | 4  |
| <b>III</b>        | <b>Results</b>                          | 4  |
| III-A             | Methodology . . . . .                   | 4  |
| III-A1            | Data preparation . . . . .              | 4  |
| III-A2            | Supervised learning . . . . .           | 5  |
| III-A3            | Unsupervised learning . . . . .         | 6  |
| III-B             | Web application for AutoML . . . . .    | 6  |
| III-C             | Outcomes . . . . .                      | 7  |
| III-C1            | Data preparation . . . . .              | 9  |
| III-C2            | Supervised learning . . . . .           | 9  |
| III-C3            | Unsupervised learning . . . . .         | 9  |
| III-D             | Discussion . . . . .                    | 11 |
| <b>IV</b>         | <b>Conclusions and Future Work</b>      | 11 |
| <b>References</b> |   | 12 |

# Automated machine learning pipeline for geochemical analysis

Oscar A. Esteban, Germán H. Alférez, Benjamin L. Clausen and Ana Martínez Ardila

*School of Engineering and Technology, Montemorelos University, Mexico*

## I. INTRODUCTION

### A. Background

The application of machine learning in geoscience has a history of around 70 years [1]. Machine learning can be defined as the ability of computers to recognize patterns without being explicitly programmed. Nowadays, different authors have proposed machine learning approaches in remote sensing [2], rock classification and predictions [3], mineral identification [4], and more. Nevertheless, the applicability of machine learning has limitations because it is necessary to have logical reasoning, programming experience, knowledge of algebra, statistics and calculus, and so on [5]. Furthermore, the creation of a highly efficient machine learning model is a time-consuming process, as it involves: cleaning and preparing the data, testing different algorithms, finding the most optimal hyperparameters, and evaluating the models with the appropriate metrics [6].

As a solution, automated machine learning (autoML) is a growing trend to face the complexity of machine learning to end users. AutoML consists of automating machine learning processes to reduce the time to create machine learning models. Also, it allows rapid implementation, and makes machine learning techniques more accessible without advanced programming knowledge [7].

### B. Problem statement

Although discrimination diagrams have been widely used since 1973 by geoscientists to classify rock samples [8], they present limitations such as: 1) leaving out important elements of the samples, so the results may be limited; 2) the overlap between the data when plotting; 3) they are effective only for a specific type of rock and even then, they can produce misclassifications; 4) discriminating the same sample with different diagrams can give different results; 5) the samples must meet certain requirements in their composition, even if the diagram indicated for their type is used; and 6) some plots are created with samples from a specific region, so their use cannot be generalized [9], [10].

There are several free and open-source software that can help to interpret geochemical-related data. For example, GCD-kit<sup>1</sup> and Igpet<sup>2</sup> allow to upload rock samples data files and discriminate them using many traditional diagrams. However, these tools are not focused on machine learning.

### C. Justification

Despite the growing interest in using machine learning approaches in geoscience, it is considered a difficult skill to learn nowadays. It is because machine learning requires mathematical and logical skills, creativity to solve problems, and knowledge to deploy the models. AutoML is a growing trend to try to avoid this issue [11]. AutoML platforms such as H<sub>2</sub>O<sup>3</sup>, DataRobot<sup>4</sup>, and Cloud AutoML<sup>5</sup> allow the automatic creation and evaluation of machine learning models. However, they are not free and some may still require prior programming knowledge.

In order to facilitate the application of machine learning to geoscience, we propose: 1) a methodology that follows a pipeline to perform machine learning techniques on geochemical datasets, and 2) a free and accessible web application that applies this methodology to the data entered using autoML. As an online tool, it does not require any installation process and can be accessed openly through a web browser.

The proposed methodology and the web application were used to analyze a granitic rock sample dataset from Southern California.

### D. Objectives

The contribution of this work is twofold. First, we present a methodology that follows a simple pipeline, allowing geoscientists to perform machine learning analysis on geochemical data. The methodology consists of: data preparation and analysis with supervised and unsupervised learning. First, the methodology starts with data preparation using techniques such as data imputation and upsampling. Then, supervised learning can be used for classifying samples with 5 classification algorithms: K-Nearest Neighbors, Decision Trees, Support Vector Machines, Logistic Regression, and Multilayer Perceptron. Unsupervised learning can be used for finding patterns with Principal Component Analysis (PCA) and clustering. Our second contribution is a web application that applies the proposed methodology allowing geoscientists to analyze geochemical data with autoML. This web application does not require programming. The methodology and the web application were used to analyze a dataset of granitic rocks from Southern California using supervised and unsupervised learning.

The specific objectives of this research work are as follows:

<sup>3</sup><https://www.h2o.ai/solutions/>

<sup>4</sup><https://www.datarobot.com/>

<sup>5</sup><https://cloud.google.com/automl/>

<sup>1</sup><http://www.gcdkit.org/>

<sup>2</sup><https://www.rockware.com/product/igpet/>

- Propose a methodology that applies: 1) data preparation techniques such as imputation and upsampling to the data before analysis; 2) supervised learning using 5 classification algorithms: K-Nearest Neighbors, Decision Trees, Support Vector Machines, Logistic Regression, and Multilayer Perceptron; 3) unsupervised learning using the PCA technique and the K-Means clustering algorithm.
- Create a web application in Python that follows the proposed methodology to perform analysis with autoML.
- Analyze a dataset with granitic rock samples from Southern California with supervised and unsupervised learning using the web application.

#### E. Hypothesis

The application of autoML to geochemical data can make machine learning techniques accessible to geoscientists.

## II. THEORETICAL FOUNDATION

### A. Underpinnings of our approach

Our approach is based on the following concepts (see Figure 1):

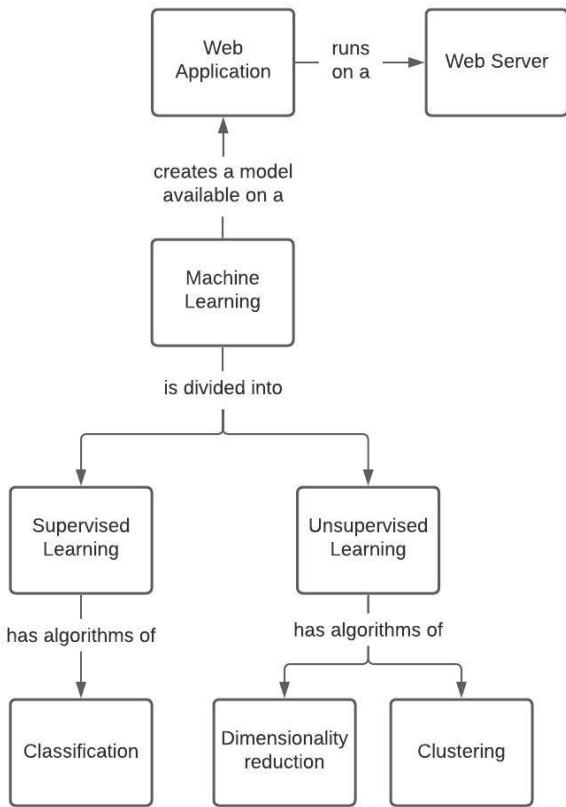


Fig. 1. Underpinnings of our approach.

- 1) Machine learning: Machine learning is a branch of artificial intelligence that allows computers to apply different techniques, learning from past experience [12]. Thanks to machine learning, the computer learns without being explicitly programmed. Machine learning involves

different areas such as computer science, engineering, statistics, data mining, and pattern recognition. Its two most used techniques are supervised and unsupervised learning [13], [14]. In supervised learning, the data is labeled. A label is a category or class to which the sample belongs and identifies it. Multiple samples can belong to the same class (have the same label). The algorithms learn from the features of each sample and predict new samples. The features are the measurable properties of the samples. Depending on the feature values (independent variables) it will be the class (dependent variable) to which it belongs. In unsupervised learning, the data is not necessarily labeled. The algorithms group the samples into clusters to discover patterns in them.

- 2) Automated machine learning: In machine learning, experience is necessary to program, train, and choose algorithms. AutoML automates these steps. Thanks to autoML, machine learning results can be obtained without having advanced technical expertise in the area of programming [15].
- 3) Supervised learning: The following algorithms are the most commonly used for classification [16]:
  - a) K-Nearest Neighbors: In this algorithm all available cases are stored. New samples are classified based on the most frequent label of its  $k$  nearest neighbors (the cases with the data most similar to it) [17].
  - b) Logistic Regression: This algorithm models the probability of an outcome based on the individual features (independent variables). The features are multiplied by a weight and then added. This sum is put into a logit function, and returns a result, that can be taken as a probability estimate [18].
  - c) Decision Trees: This algorithm splits the data into subsets (creating a tree) based on the most important features that make the set distinct. It has *decision blocks* and *terminating blocks*. In decision blocks, there are two alternatives, depending on whether the condition is true or false. They can lead to another decision block or a terminating block. In a terminating block some conclusion has already been reached. Each block has a measure called *entropy*. Entropy measures the disorder or uncertainty in a group of samples. The higher the entropy, the messier the data is. The Decision Trees algorithm tries to decrease this measure as each block progresses. When a conclusion is reached (in a terminating block), the entropy is 0 [19].
  - d) Support Vector Machines: In this algorithm, the data is plotted in a  $n$ -dimensional space (number of features) and a decision boundary (or hyperplane) split it into classes. The further the plotted data points are from the decision boundary, the more confident the algorithm is about the prediction. The data points closest to the hyperplane are called *support vectors* [20].
  - e) Multilayer Perceptron: It is the most common artificial neural network (ANN). It has 2 layers directly connected to the environment (*input layer* and *output*

*layer).* The intermediate layers between these two are called *hidden layers*. Each layer contains neurons connected to the next layer.

The signal transmitted by neurons follows a single direction (from the input layer to the output layer) without forming loops. This structure is called *feed forward*. It also uses an algorithm called *backpropagation* to minimize errors between the model outputs and the expected outputs [21].

- 4) **Unsupervised learning:** Data reduction and clustering are commonly used together in unsupervised learning to improve accuracy by reducing the dimensions of the data [22]. In this research work, the following 2 unsupervised learning methods are used:
  - a) **PCA:** It is a technique used to reduce the dimensionality of the data while losing as little information as possible. The dataset is transformed to a coordinate system. A first axis is chosen in the direction of the most variance in the data. A second axis is chosen orthogonal to the first axis and with the largest variance that it can. This process is repeated until all the data is covered on the generated principal components (PCs) [23].
  - b) **K-Means:** It is an algorithm to form clusters with similar characteristics. The center of each cluster is called *centroid*, and it is the mean distance of the values in each cluster. The K-Means algorithm finds  $k$  unique clusters and each sample is assigned to the cluster with the closest centroid [24].
- 5) **Web Application:** A web application is any tool that is hosted on a web server and which can be accessed via a web browser. Its functions can be any type and be very simple to very complex. Because it is hosted on a server, it is not necessary to install the application on a computer. Rather, it interacts with the data from the web, creating a *client-server* environment. Any application used to enter the information is called *client*. The *server* can be any hardware or software that uses different protocols to respond to client requests.

## B. Related Work

Several authors have proposed the application of machine learning techniques to solve problems in geochemistry. Table 1 summarizes these research works.

1) *Supervised learning:* In the following research works, the authors compare several supervised learning approaches to create optimal classifiers and predict new samples.

In [25], the authors proposed an approach with Multinomial Logistic Regression to discriminate the source rock of detrital monazites. They used 16 elements (La, Cr, Pr, Nd, etc.) from samples of detrital monazites from African rivers. All possible combinations were created using the 16 elements (65,535 different combinations) to obtain the models with the best discrimination. Accuracies by number of elements were compared and the results showed that the highest accuracy (97%) is obtained with 8 to 10 elements.

In [26], the authors applied 3 classification algorithms (Random Forest, Gradient Boosted Machine, and Deep Neural Network) to predict altered and non-altered lithotypes. They used a dataset with geophysical log data from 1,230 coal samples taken from 263 boreholes from the Leichhardt Seam of the Bowen Basin in Eastern Australia. The dataset was randomly split into an 80% training set and 20% testing set. The Random Forest model performed the best with average results of: 99% precision, 99% recall, and 99% F-score for the training set; and 97% precision, 93% recall, and 95% F-score for the testing set. Only 11 classifications out of 1,230 samples were wrong.

In [27], the authors compared and discarded different approaches (Discriminant Analysis, Logistic Regression Analysis, Decision Trees, etc.) to develop an accurate protolith classifier. A dataset was created and normalized extracting 9 major elements ( $\text{SiO}_2$ ,  $\text{TiO}_2$ ,  $\text{Al}_2\text{O}_3$ ,  $\text{MgO}$ , etc.) from 533,360 samples: 497,401 igneous samples and 35,959 sedimentary samples. The samples were taken from a global dataset of rock major elements. The results showed that the best classifier was an Ensemble Trees model (RUSboost) with an accuracy of 95% true igneous and 85% true sedimentary.

In [28], the authors compared 3 classification algorithms (Support Vector Machines, Random Forest, and Sparse Multinomial Regression) for the discrimination of volcanic rocks according to 8 tectonical settings. The dataset was obtained from 2 global geochemical databases: PetDB and GEOROC. It was composed of 24 geochemical data and 5 isotopic ratios (29 features) and contained 2,074 samples. The results showed that the 3 methods presented an accuracy higher than 83% in most of the classes. Although the accuracy of Sparse Multinomial Regression was the lowest, it was the most useful method for generating geochemical signatures that were easy to interpret and analyze.

In [29], the authors used Support Vector Machines to classify rock samples according to 8 different tectonical settings. The dataset was composed of major elements, trace elements, and isotopes, from 3,095 samples. They classified the samples using major elements, trace elements and isotopes separately, and the combination of all (4 experiments total). The results showed that the combination of the major elements, trace elements and isotopic data offers the highest accuracy (93%) than separately: 79% for major elements, 87% for trace elements, and 79% for isotopes.

2) *Unsupervised learning:* In the following research works, unsupervised learning was used to find patterns in geochemistry data. They also show the relationship between data reduction and clustering.

In [30], the authors applied a clustering method based on a hierarchy to interpret geochemical data from the soil of Colorado, USA. The dataset was cleaned based on the concentration percentage of the elements, and PCA. The final dataset contained 959 samples with 22 PCs. The results of the hierarchy method were 2 clusters, each one with elements in common. Cluster 1 contained elements commonly enriched in shales and other fine-grained marine sedimentary rocks. Cluster 2 contained elements commonly associated with potassium feldspars or felsic rocks. The plotted results were consistent

TABLE I  
RESEARCH WORKS THAT APPLY MACHINE LEARNING IN THE AREA OF GEOSCIENCE

| Authors   | Year | Features (independent variables)  | Classes (dependent variables)  | ML Approach   |
|---|------|---|--|---|
| Itano, K.; Ueki, K.; Izuka, T. & Kuwatani, T. [25]          | 2020 | Detrital monazite elements (La, Ce, Pr, Nd, Sm, Eu, etc.)   | Source rock types (garnet-free, garnet-bearing, or granitic rocks)         | Multinomial Logistic Regression   |
| Maxwell, K.; Rajabi, M. & Esterle, J. [26]                  | 2019 | Geophysical log data  | Lithotypes (altered or non-altered)  | Random Forest, Gradient Boosted Machine, and Deep Neural Network          |
| Hasterok, D.; Gard, M.; Bishop, C. & Kelsey, D. [27]        | 2019 | Rock major elements ( $\text{SiO}_2$ , $\text{TiO}_2$ , $\text{Al}_2\text{O}_3$ , $\text{MgO}$ , etc.)  | Protoliths (igneous or sedimentary)  | K-Nearest Neighbors, Logistic Regression, Decision Trees, etc.            |
| Ueki, K.; Hino, H. & Kuwatani, T. [28]                      | 2018 | Geochemical data ( $\text{SiO}_2$ , $\text{TiO}_2$ , $\text{Fe}_2\text{O}_3$ , etc.) and isotopic ratios ( $^{206}\text{Pb}/^{204}\text{Pb}$ , $^{207}\text{Pb}/^{204}\text{Pb}$ , etc.)                                | Tectonical settings (continental arc, continental flood, island arc, etc.) | Support Vector Machines, Random Forest, and Sparse Multinomial Regression |
| Petrelli, M. & Perugini, D. [29]                            | 2016 | Major elements ( $\text{SiO}_2$ , $\text{TiO}_2$ , $\text{Al}_2\text{O}_3$ , etc.), trace elements (Sr, Ba, Rb, Zr, etc.), and isotopes ( $^{206}\text{Pb}/^{204}\text{Pb}$ , $^{207}\text{Pb}/^{204}\text{Pb}$ , etc.) | Tectonical settings (continental arc, island arc, intra-oceanic arc, etc.) | Support Vector Machines   |
| Ellefse, K. J. & Smith, D. B. [30]                          | 2016 | Geochemical data (As, Cd, Sb, S, K, Ba, Th, Na, etc.)   | No (clustering)  | Hierarchical Clustering   |
| Jiang, Y.; Guo, H.; Jia, Y.; Cao, Y. & Hu, C. [31]          | 2015 | Geochemical data (Ca, Cl, Na, $\text{NO}_3$ , pH, etc.)   | No (clustering)  | PCA and Hierarchical Cluster Analysis                                     |
| Alférez, G. H.; Rodriguez, J.; Clausen, B. & Pompe, L. [32] | 2015 | Geochemical data ( $\text{SiO}_2$ , Sr, Gd/Yb, and $\text{K}_2\text{O}/\text{SiO}_2$ )  | No (clustering)  | K-Means   |

with the map of geological units in the area.

In [31], the authors applied PCA and Hierarchical Cluster Analysis (HCA) to study the geochemical processes that control the presence of As in groundwater in the Hetao basin, Mongolia. 90 groundwater samples with 22 geochemical parameters (Ca, Cl, Na,  $\text{NO}_3$ , pH, etc.) were collected from the area. PCA was applied to the samples and they identified 4 major PCs that explain 78.2% variance of the original data. The components were the input for the HCA method. The results showed 3 clusters. In Cluster 1, high As concentrations correspond to high P concentrations in flat plain. In Cluster 2, samples are affected by lithological and redox factors. In Cluster 3, low As concentrations correspond to low P concentrations in alluvial fans.

In [32], we compared PCA and Geographic Information Systems (GIS) techniques with the K-Means clustering algorithm. We used geochemical data from 800 rock samples from an area of Southern California. The approaches were compared in terms of 4 geochemical factors:  $\text{SiO}_2$ , Sr, Gd/Yb, and  $\text{K}_2\text{O}/\text{SiO}_2$ . The results showed that the K-Means algorithm gives results very similar to the ones obtained with GIS and PCA.

3) *Discussion:* According to the research works presented above, supervised learning is more used than unsupervised learning in geochemistry. In general, research works on supervised learning use the following methodology: cleaning and selecting the data, splitting the dataset, training the algorithms, and showing the results. However, several of these research works lack specific activities for data preparation, such as imputation, removing null values, or sample balancing. Also, they tend to leave out other metrics besides accuracy to evaluate and compare the models. There are not research works

that propose a machine learning platform or an autoML tool for geochemical analysis.

### III. RESULTS

#### A. Methodology

The methodology proposed for applying machine learning to geochemical data is as follows (see Figure 2). First, the data is entered and prepared using imputation or removing null values. Then, the learning method to be applied is chosen. For supervised learning: 1) the class and features are selected; 2) the user can choose to balance the data or not; 3) the dataset is split; 4) the models generated are automatically trained and evaluated; and 5) the best model can be used to classify new samples. For unsupervised learning: 1) the features are selected; 2) PCA is applied; and 3) K-Means clustering is applied. These steps are explained as follows.

1) *Data preparation:* In this step, the dataset is uploaded and prepared. The user can choose to apply imputation to fill null values. Otherwise, the user can choose to remove samples with null values. Specifically, the following activities are carried out during data preparation:

- 1) Upload the dataset: The user uploads the dataset to the server. Our web application is publicly available<sup>6</sup>. First, the file must be in the comma separated values (CSV) format. The CSV format is widely used to store raw data. Each column must be identified with a name. Also, the CSV dataset must only contain numeric or string data values to avoid problems when reading special characters. For example, there are not allowed characters like <, >, %, ^, etc. For supervised learning, samples must be labeled

<sup>6</sup><http://201.134.41.15:8889/>

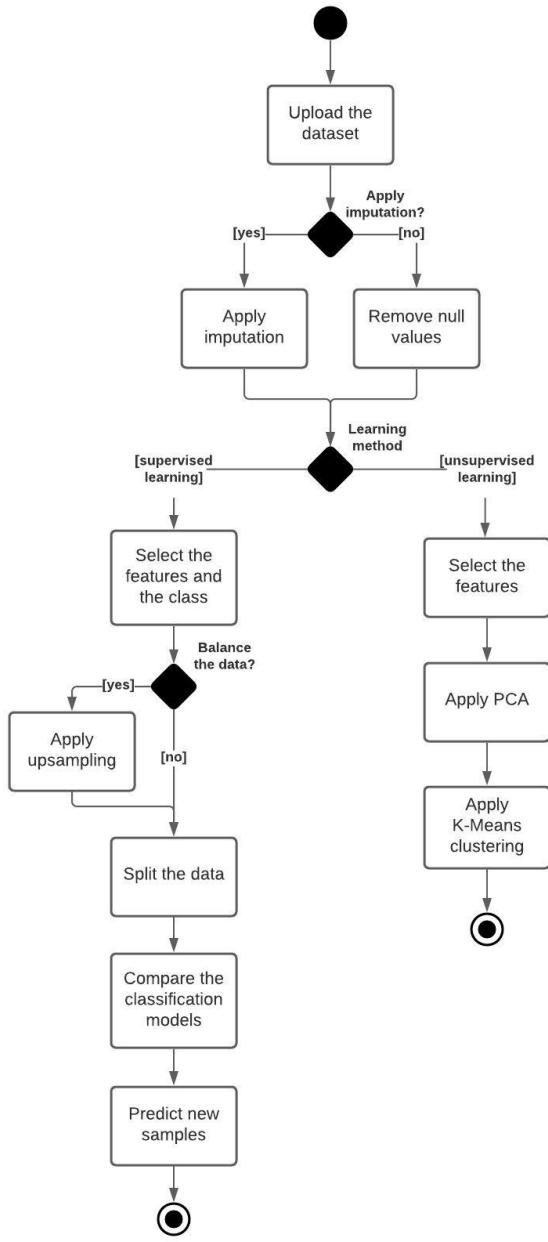


Fig. 2. Pipeline to apply machine learning to geochemical data with autoML.

by strings of characters or categorical values (integer natural numbers). For unsupervised learning, it is not necessary to have labeled samples.

- 2) **Apply imputation:** If the dataset has samples with missing values, the user can decide to use imputation or not. If the user decides not to use imputation, then the samples with null values are removed. Otherwise, imputation is applied. Imputation refers to filling empty spaces with different techniques. The web application uses the Extra (extremely randomized) Tree Regressor to perform this technique [33]. Extra Tree Regressor is an Ensemble Decision Tree algorithm. It can produce better results than Decision Trees because it splits each node randomly instead of looking for the most optimal split.

**2) Supervised learning:** In this step, the classification algorithms are trained and evaluated. The most accurate model is then used to classify new samples of data. The activities to apply the supervised learning method are described as follows:

- 1) **Select the features and the class variables:** The class and features to train the algorithms are selected. In the web application, the class labels can be string of characters or numeric categories, and the features must be numeric values.
- 2) **Balance the data:** The upsampling method can be applied to prevent the most frequent one from dominating the algorithm. Upsampling consists of randomly duplicating samples of the least frequent class until its quantity is the same as the most frequent class. Also, the classes with which the classifiers will be trained are selected.
- 3) **Split the data:** The percentage of data used for training is selected, and the rest is used for testing. Commonly between 70% and 80% of all data is used to train the models and the rest is used to evaluate their performance.
- 4) **Compare the classification models:** The data is finally entered in the classifiers for training and evaluation. The web application uses the HPO (Hyper Parameter Optimization) technique to find the best configuration for each model. A hyper parameter is a defined variable that affects the performance of the algorithm. In HPO, several values for each hyper parameter are selected, and the model is evaluated with the possible combinations [34]. Then, the model with the best performance is automatically chosen. The following metrics are used to evaluate each model:
  - a) **Accuracy:** It represents the percentage of samples classified correctly out of total samples. It is defined by the following formula:
 
$$\text{Accuracy} = \frac{\text{Correct predictions}}{\text{Total predictions}}$$
  - b) **Precision:** It represents the percentage of samples correctly identified as positive (true positives) out of total samples identified as positive (true positives + false positives). It is defined by the following formula:
 
$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$
  - c) **Recall:** It represents the percentage of samples correctly identified as positive (true positives) out of total positive samples (true positives + false negatives). It is defined by the following formula:
 
$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$
  - d) **F-score:** It represents the harmonic mean between precision and recall. It is defined by the following formula:
 
$$\text{F-score} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$
  - e) **Feature importance:** It is an additional metric for the Decision Trees classifier. It represents the percentage of how much the model performance decreases when a feature is not available. A feature is important if shuffling its values increases the model error.
- 5) **Predict new samples:** In this step, the samples from a new dataset are classified. The new dataset must contain only

the features selected in Activity 1 (Select the features).

3) *Unsupervised learning*: In this step, the features of the dataset are reduced using the PCA technique. The PCs generated are used as input to the K-Means clustering method. The activities to apply the unsupervised learning method are described as follows:

- 1) Select the features: The features to be analyzed by PCA and clustering are selected. The features must be continuous numerical values, not categorical. Although PCA can be applied to discrete values, it is not recommended because the variance is less significant in them and the results obtained are less relevant.
- 2) PCA: The PCA technique is applied for data reduction. It aims to reduce a large number of variables to one (much) smaller number losing as little information as possible. Each component contains the combination of the original variables and the largest variance available in the data. PCA reduces data noise by grouping multivariate into fewer components.
- 3) K-Means clustering: The PCs that were obtained with PCA are used as input for the K-Means clustering algorithm. Clustering consists of grouping unlabeled samples with similarities between them. The clusters help to understand the organization of the data in a summarized way.

#### B. Web application for AutoML

The Flask<sup>7</sup> micro framework was used to create the web application for autoML. Flask was chosen in this research work because it is easier and faster to learn than other frameworks as it provides tools to define routes, manage forms, render templates, etc. while external packages can extend it [35]. Flask has two main dependencies: Werkzeug provides the routing, debugging, and Web Server Gateway Interface (WSGI) subsystems; and Jinja2 provides support for the view component. A view is the response sent by the application for a web request and each view is associated with a specific route.

In Flask, the operations with the data are performed by the model components and when finished, it redirects to a view. The view and model components of the web application were programmed in Flask using Python. The source code and the templates are available online<sup>8</sup>. The *app.py* file contains the definition of the routes, functions and views of the web application. The interactions between the user and the components in the web application are as follows (see Figure 3).

First, the user enters the web application and the route of the *Index()* function (see step 1 in Figure 3) is requested. When the route matches, the function renders the *index.html* template and displays it to the user (see step 2 in Figure 3). The HTML templates contain forms where the user chooses the operations to be applied to the entered dataset. In the *index.html* template the user enters the dataset to analyze. Once the form is submitted, the path belonging to the *Add\_CSV()*

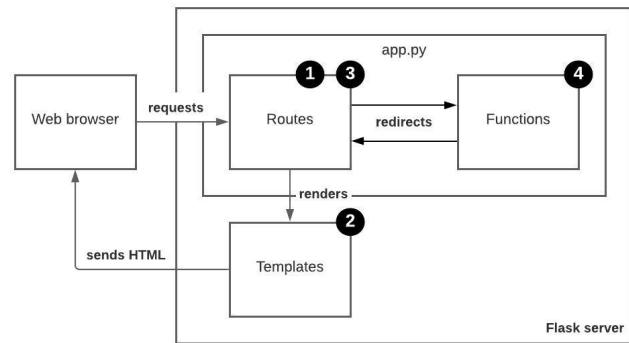


Fig. 3. Components in Flask of the web application.

function is requested (see step 3 in Figure 3). The *Add\_CSV()* function reads the CSV file and converts it to a dataframe (see step 4 in Figure 3). At the end of its operations, the function redirects to the route of the *Imputation()* function and the process repeats again from step 1 until it reaches an endpoint in the pipeline shown in Figure 2.

The underlying process followed by the web application is shown in Figure 4. White blocks are the view functions that render the HTML templates, which are the yellow blocks. Blue blocks are functions that work with the data between each view and at the end, they redirect to the next view. Each set of white, yellow and blue blocks corresponds to an activity of the methodology presented in the previous section.

The Data preparation step is composed of 3 view functions, 3 HTML templates and 3 work functions. The *Add\_CSV()* function transforms the CSV file to a dataframe; the *Add\_Imput()* function applies imputation to the data or removes the null values according to the user's response; and the *Choose\_Learning()* function redirects to the next step, depending on the analysis that you want to apply to the dataset.

The Supervised learning step is composed of 6 view functions, 6 HTML templates and 6 work functions. The *Add\_SupFeats()* function keeps only the columns selected by the user; the *Add\_Balance()* function applies upsampling or not, according to the user's response; the *Add\_Split()* function splits the dataset to train and evaluate the 5 algorithms, and chooses the most accurate one; the *Add\_Report()* function redirects to the template to enter the new dataset to be classified; the *Add\_Classify()* function classifies the new samples; and the *Add\_Results()* function downloads a CSV file with the samples and their predicted label.

The Unsupervised learning step is composed of 3 view functions, 3 HTML templates and 3 work functions. The *Add\_UnsupFeats()* function applies PCA to the columns selected by the user; the *Add\_PCA()* function groups the components into clusters; and the *Add\_Results()* function allows to download the original samples and their assigned cluster in a CSV file.

A session is used to store user information through requests. This functionality was necessary to avoid data lost at the end of the session. The *session* object creates a cookie to store the content of the session in a temporary directory on the server.

<sup>7</sup><https://flask.palletsprojects.com/en/1.1.x/>

<sup>8</sup><https://github.com/OscarAlex/geoml>

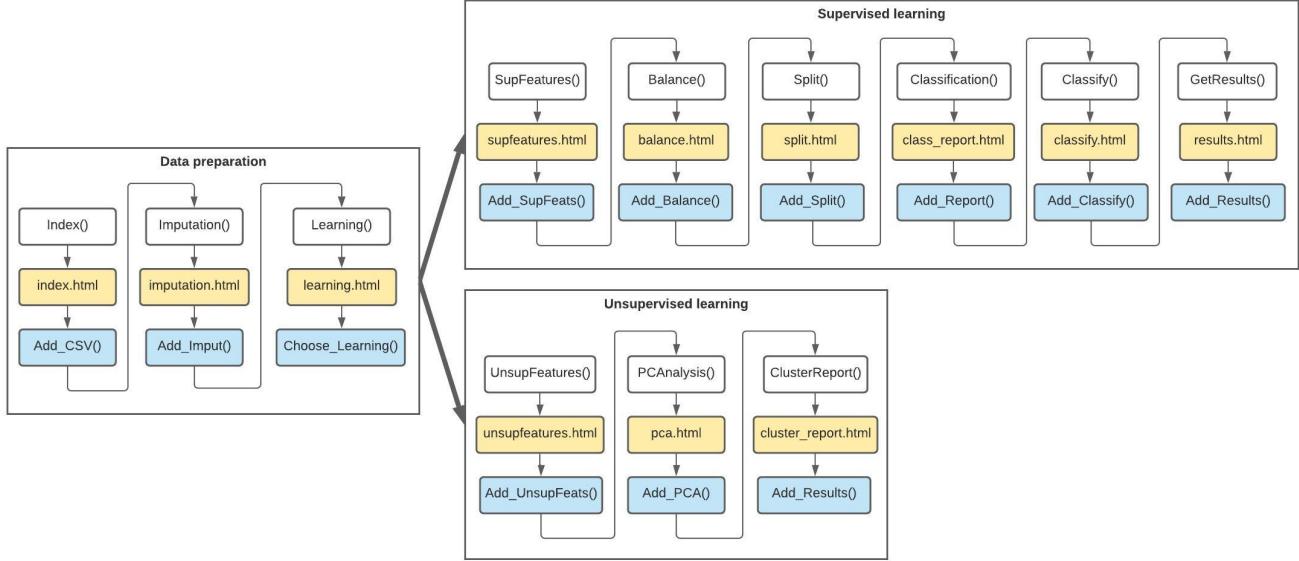


Fig. 4. Underlying process of the autoML web application.

The production WSGI (Web Server Gateway Interface) server Waitress<sup>9</sup> was used for the deployment of the web application. WSGI is a standard used to establish how the web server communicates with the web application.

### C. Outcomes

Our approach was evaluated with a compositional dataset from 6 fault-separated blocks in the Peninsular Ranges Province and Transverse Ranges Province. The Peninsular Ranges are a group of mountain ranges, stretching from Southern California to Southern Baja California, Mexico. North of the Peninsular Ranges Province is the east-west Transverse Ranges Province. Around this area, there are several faults that allow the subdivision of the provinces: San Andreas, San Jacinto, Elsinore, Pinto Mountain, and Banning. There are 6 structurally-bounded units or blocks bounded by these faults: San Gabriel, San Bernardino, and Little San Bernardino from Transverse Ranges; and San Jacinto, Perris, and Santa Ana from Peninsular Ranges [36].

The most important geological feature of the Peninsular Ranges Batholith (PRB) is a batholith-wide separation into western and eastern parts based on geophysical criteria. The older western terrain is more mafic and heterogeneous, and plutons were generally emplaced at a shallower depth with a shallower magma source than those in the east. The magmatism in these provinces records a west to east progression of subduction transitioning from an oceanic to a continental arc setting characterized by numerous individual plutons with compositions ranging from gabbro to tonalite [37].

The dataset was composed using 514 granitic rock samples (quartz diorites, granodiorites, and quartz monzonites) collected from the study area. This dataset contains 8 major elements ( $\text{SiO}_2$ ,  $\text{Al}_2\text{O}_3$ ,  $\text{Fe}_2\text{O}_3$ ,  $\text{MgO}$ ,  $\text{CaO}$ ,  $\text{Na}_2\text{O}$ ,  $\text{K}_2\text{O}$ , and

$\text{TiO}_2$ ), 36 trace elements ( $\text{P}_2\text{O}_5$ ,  $\text{MnO}$ ,  $\text{Sc}$ ,  $\text{V}$ ,  $\text{Cr}$ ,  $\text{Mn}$ ,  $\text{Co}$ ,  $\text{Ni}$ ,  $\text{Cu}$ ,  $\text{Zn}$ ,  $\text{Rb}$ ,  $\text{Sr}$ ,  $\text{Y}$ ,  $\text{Zr}$ ,  $\text{Nb}$ ,  $\text{Mo}$ ,  $\text{Cs}$ ,  $\text{Ba}$ ,  $\text{La}$ ,  $\text{Ce}$ ,  $\text{Pr}$ ,  $\text{Nd}$ ,  $\text{Sm}$ ,  $\text{Eu}$ ,  $\text{Gd}$ ,  $\text{Tb}$ ,  $\text{Dy}$ ,  $\text{Ho}$ ,  $\text{Er}$ ,  $\text{Tm}$ ,  $\text{Yb}$ ,  $\text{Hf}$ ,  $\text{Ta}$ ,  $\text{W}$ ,  $\text{Th}$ , and  $\text{U}$ ), sample id, latitude, longitude, and block (as the class) from the collected samples. Two datasets were created from the original dataset. The first dataset was composed by 90% of the samples per block, and the remaining 10% of the samples were used for the second dataset. The first dataset (462 samples) was splitted again for the training and testing of the models (80% for training and 20% for testing). The labels of the second dataset (52 samples) were removed and the samples were used for prediction. Both datasets were used for supervised learning. The original dataset (514 samples) was used for unsupervised learning. The classes were not used for PCA and clustering. Table 2 shows the number of samples per block for the 3 datasets.

TABLE II  
SAMPLES PER CLASS OF EACH DATASET

| Block                 | Original dataset | Training and testing dataset | Prediction dataset |
|-----------------------|------------------|------------------------------|--------------------|
| San Bernardino        | 129              | 116                          | 13                 |
| Perris                | 126              | 113                          | 13                 |
| San Jacinto           | 99               | 89                           | 10                 |
| Santa Ana             | 91               | 82                           | 9                  |
| San Gabriel           | 45               | 41                           | 4                  |
| Little San Bernardino | 24               | 21                           | 3                  |
| <b>Total</b>          | <b>514</b>       | <b>462</b>                   | <b>52</b>          |

For both analyzes, the samples with null values were removed in the Data Preparation step. Figure 5 shows the workflow of the web application. Each screen shows a form where the user chooses the operations to perform on the dataset. The samples datasets, the evaluation results, and the resulting plot images of the analysis are available online<sup>10</sup>.

<sup>10</sup><https://github.com/OscarAlex/Automated-machine-learning-pipeline-for-geochemical-analysis>

<sup>9</sup><https://docs.pylonsproject.org/projects/waitress/en/stable/>

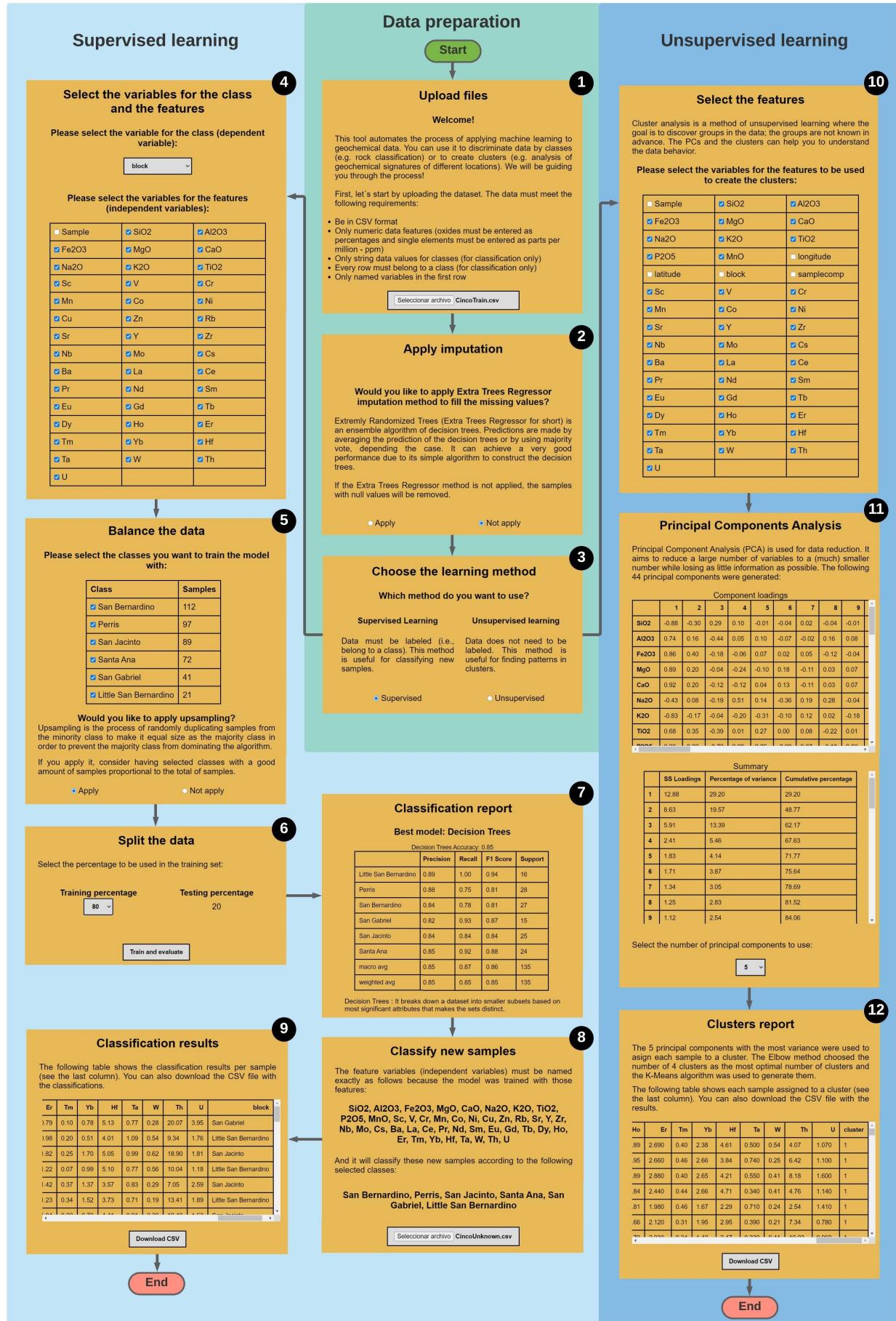


Fig. 5. Workflow of the web application.

1) *Data preparation:* Data was prepared by eliminating the rock samples with null values. The activities carried out to prepare the data are as follows:

- 1) Upload the dataset: Step 1 in Figure 5 shows a screenshot of the interface to enter the dataset. In this case, the training dataset was uploaded to the web application.
- 2) Apply imputation: Step 2 in Figure 5 shows a screenshot of the interface to apply or not apply imputation to the dataset. In this example, imputation was not applied, so samples with null values were removed. After removing null values, the remaining dataset was composed of 432 samples. The remaining samples per class were as follows: San Bernardino: 112 samples; Perris: 97 samples; San Jacinto: 89 samples; Santa Ana: 72 samples; San Gabriel: 41 samples; Little San Bernardino: 21 samples.
- 3) Choose the learning method: Step 3 in Figure 5 shows a screenshot of the interface to select the learning method to be applied to the dataset. Supervised learning was applied first and then unsupervised learning.

2) *Supervised learning:* Supervised learning was applied to classify granitic samples according to the 6 blocks of the Peninsular Ranges and Transverse Ranges region: San Gabriel, San Bernardino, Little San Bernardino, San Jacinto, Perris, and Santa Ana. The activities carried out in terms of supervised learning were as follows:

- 1) Select the features and the class variables: The class column and the feature columns were selected (see step 4 in Figure 5). The column *block* was selected as the class. The major and trace elements of the samples (44 out of 48 features) were also selected. The following features were left out: *latitude*, *longitude*, and *sample id*.
- 2) Balance the data: In this step, the data was balanced (see step 5 in Figure 5). The class with the highest frequency was *San Bernardino* with 112 samples, and the lowest, *Little San Bernardino* with 21 samples. Upsampling was applied to balance the frequency of the 6 classes. After applying upsampling, the remaining dataset was composed of 672 samples, 112 samples per class.
- 3) Split the data: The percentage values to split the dataset were selected (see step 6 in Figure 5). Specifically, 80% of the samples were selected to train the algorithms and the remaining 20% of the samples were used evaluate the resulting models.
- 4) Compare the classification models: In this step, the web application returns the metrics of the models (see step 7 in Figure 5). The model with the best performance in this example was the one generated with the Decision Trees algorithm, with an accuracy of 87%. Its precision, recall and F-score values were also good in general (see Table 3). The classes that the model best classified were *San Gabriel* and *Santa Ana*, both with 95% in F-score. Contrary, *San Bernardino* was the class with the lowest recall (68%) and F-score (79%). The 10 most important features of the Decision Trees model and their values are shown in Table 4. Ni and Sc may be the most important as they are mafic. La, Sr, Y, and Tb are important because of pressure effects while Cs and K<sub>2</sub>O are important because

TABLE III  
REPORT FROM THE DECISION TREES MODEL

| Block                 | Precision | Recall | F-score |
|-----------------------|-----------|--------|---------|
| Little San Bernardino | 84%       | 100%   | 91%     |
| Perris                | 87%       | 91%    | 89%     |
| San Bernardino        | 95%       | 68%    | 79%     |
| San Gabriel           | 91%       | 100%   | 95%     |
| San Jacinto           | 77%       | 85%    | 81%     |
| Santa Ana             | 100%      | 90%    | 95%     |

TABLE IV  
FEATURE IMPORTANCE OF THE DECISION TREES MODEL

| Feature                        | Importance |
|--------------------------------|------------|
| Ni                             | 0.1874     |
| Sc                             | 0.1215     |
| Cu                             | 0.0977     |
| La                             | 0.0903     |
| Sr                             | 0.0633     |
| Cs                             | 0.0561     |
| Y                              | 0.0479     |
| Fe <sub>2</sub> O <sub>3</sub> | 0.0422     |
| Tb                             | 0.0305     |
| K <sub>2</sub> O               | 0.0294     |

they are usually associated with magma depth source [38].

Figure 6 shows the shortest path to reach a conclusion in the Decision Trees model. It is explained as follows. Each block of the tree contains: its entropy value, its condition and the number of samples that meet it, a one-dimensional array that indicates the score value of each class, and the class with the highest score value. When a new sample is introduced to be classified, it is located in the first block. If the condition is met in the first block (Ni <= 0.006), it goes to the decision block on the left and the sample is classified as *San Jacinto*. If the condition is met in the second block (Cu <= 0.007), it goes to the left block and the sample is classified as *Perris*. If the condition is met in the third block (Cs <= 0.001), it goes to the left block and the sample is classified as *San Gabriel*, and so on until reach a conclusion. If the condition is false in this block, it goes to the right and the sample is finally classified as *Perris* in the terminating block.

The accuracy of the remaining models was: K-Nearest Neighbors: 85%; Logistic Regression: 41%; Support Vector Machines: 43%; and Multilayer Perceptron: 77%.

- 5) Classify new samples: The testing dataset was entered to be classified by the Decision Trees model (see step 8 in Figure 5). The testing dataset contained the major and trace elements selected by the user. Step 9 in Figure 5 shows the table with the features and the predicted label for each sample. The table with the sample features and its predicted class can be downloaded in CSV format.

3) *Unsupervised learning:* Unsupervised learning was applied to observe the behavior of clusters in terms of geochemical elements. The activities carried out in terms of unsupervised learning were as follows:

- 1) Select the features: The columns with the features were

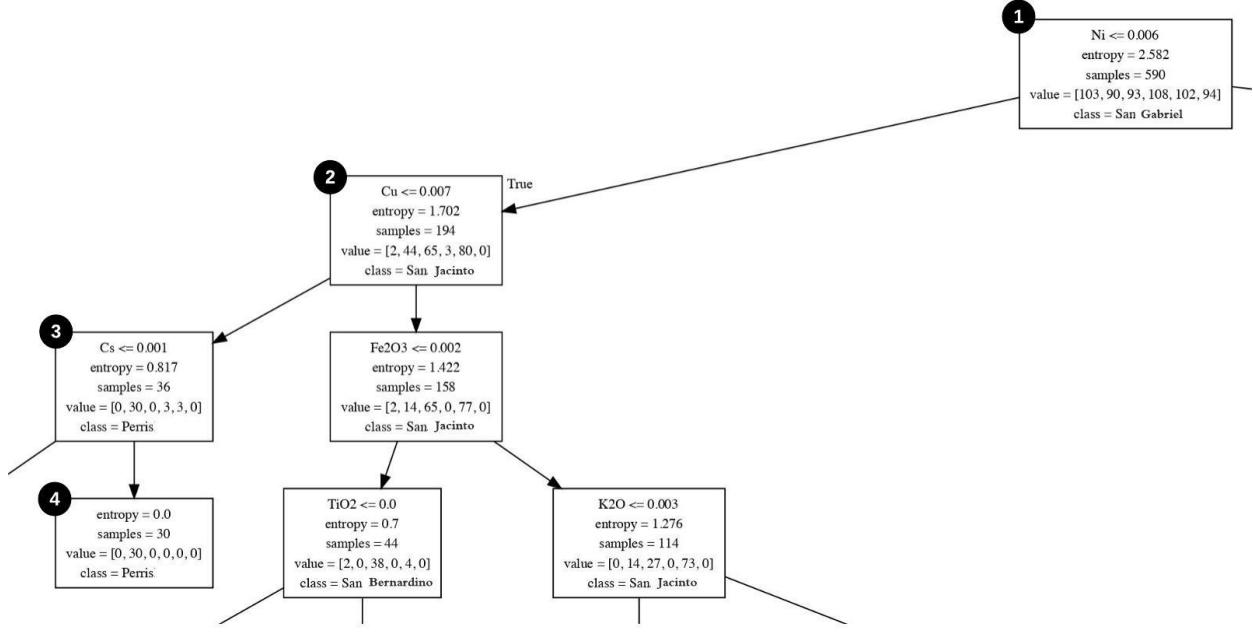


Fig. 6. Shortest path in the decision tree generated.

selected (see step 10 in Figure 5). The major and trace elements of the samples (44 out of 48 features) were selected. The following features were left out: *sample id*, *latitude*, *longitude*, and *block*.

- 2) PCA: PCA was applied to reduce noise in the data and improve the performance of the K-Means algorithm. Step 11 in Figure 5 shows the screenshot with the table containing the features, and the calculated component loadings. Table 5 shows the eigenvalues, variance percentage, and cumulative variance percentage of the PCs generated. The 5 first PCs were selected to explain 71.9% of the data variance.

TABLE V  
EIGENVALUES AND VARIANCE OF THE GENERATED PCs

| PC   | Eigenvalue | Variance percentage | Cumulative variance percentage |
|------|------------|---------------------|--------------------------------|
| PC1  | 12.93      | 29.32%              | 29.32%                         |
| PC2  | 8.58       | 19.45%              | 48.77%                         |
| PC3  | 6.00       | 13.61%              | 62.39%                         |
| PC4  | 2.41       | 5.46%               | 67.85%                         |
| PC5  | 1.79       | 4.05%               | 71.9%                          |
| PC6  | 1.72       | 3.91%               | 75.81%                         |
| PC7  | 1.33       | 3.01%               | 78.82%                         |
| PC8  | 1.25       | 2.83%               | 81.65%                         |
| PC9  | 1.09       | 2.47%               | 84.13%                         |
| PC10 | 0.96       | 2.18%               | 86.31%                         |

- 3) K-Means: Cluster analysis with K-Means was applied to observe the relationship between the geochemical elements of the samples. Step 12 in Figure 5 shows a screenshot with the table that contains each sample and its corresponding cluster. The table with the sample features and its assigned cluster can be downloaded in CSV format. According to the Elbow method, 3 was

chosen as the optimal value of *k* (number of clusters). Figure 7 shows the sample clusters plotted by longitude and latitude. The average SiO<sub>2</sub> values are as follows: Cluster 1 = felsic @ SiO<sub>2</sub> average = 71%; Cluster 2 = intermediate @ SiO<sub>2</sub> average = 65%; and Cluster 3 = mafic @ SiO<sub>2</sub> average = 58%. However, the average SiO<sub>2</sub> values are not quite right to define the compositional groups (for example mafic compositions are not usually higher than 55% SiO<sub>2</sub>).

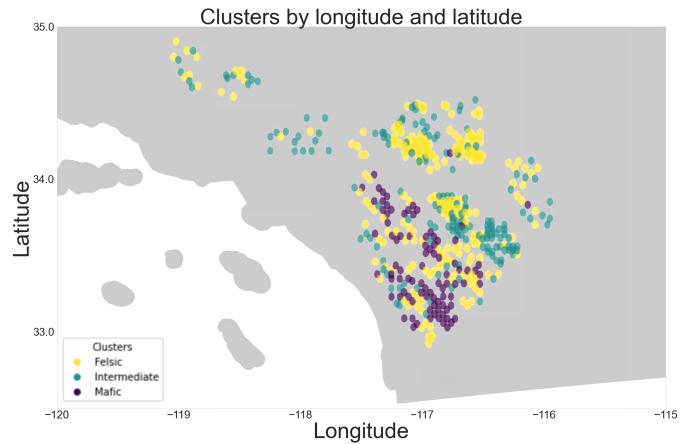


Fig. 7. Cluster map of the Transverse Ranges Province and the Peninsular Ranges Province in Southern California. The samples were located according to their measured latitude and longitude.

The clusters were generated in terms of the samples and geochemical elements. Figure 8 shows the clusters plotted in terms of PC1 and PC2. Positively correlated variables were grouped together (for example: MnO, Mn, and TiO<sub>2</sub> are positively correlated). Negatively correlated

variables were placed on opposite quadrants of the plot origin (for example:  $\text{SiO}_2$  is negatively correlated to  $\text{Fe}_2\text{O}_3$ ). The distance between the variables from the plot origin measures the quality of the variables on the factor map (for example: Gd and Dy are more represented in PC2 than in PC1, and Cu and Ni are not well represented in both PCs). PC1 is probably related to compatible/incompatible elements because  $\text{K}_2\text{O}$ ,  $\text{SiO}_2$ , and Rb were heavily represented on it.

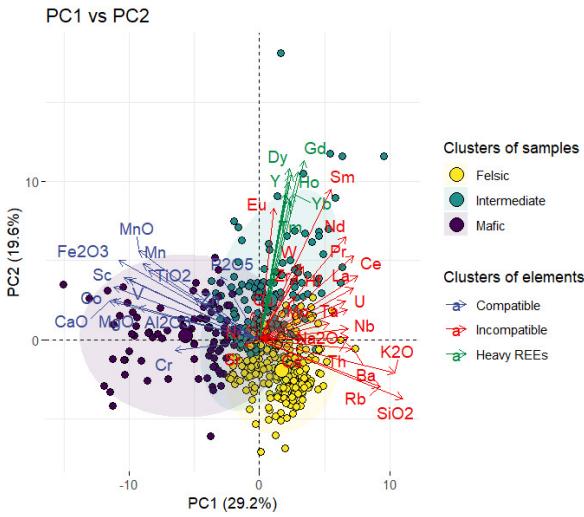


Fig. 8. Clusters according to PC1 and PC2. For the element clusters: red are large ionic radius, blue are small ionic radius, and green are Rare Earth Elements (REEs).

Clusters plotted on PC2 and PC3 show the effects of pressure (see Figure 9). Sr is positive and Y is negative to PC3 as expected if PC3 is related to pressure. The other Rare Earth Elements (REEs) arrange themselves in between. PC2 does not show much dispersion. However, the large ionic radius elements ( $\text{K}_2\text{O}$  and Rb) are negative, and the small ionic radius ( $\text{MgO}$ , Co, V, and Mn) are positive. For the element clusters: green seems to be heavy REEs (small ionic radius); red are light REEs (large ionic radius); and blue are compatible elements (small ionic radius). For the sample clusters: yellow is high in  $\text{SiO}_2$ ,  $\text{K}_2\text{O}$ , and Rb (felsic); purple is high in compatible elements (mafic); and green seems to be high in REEs (intermediate).

Clusters plotted on PC4 and PC5 could be related to water effects (see Figure 10). Immobile Ta and Nb are positive PC4, along with U and Th. For the element clusters: red includes the mobile alkaline elements (Na, K, Rb, and Cs) as well as the immobile ones (Nb and Ta). Perhaps it also includes the elements carried during hydrothermal alteration (Cu, Mo, and W), the radioactive elements (U, Th, K, and Rb) and the Zr-Hf set. For the sample clusters: seem to all center on zero, so the extent of fractionation is not related to water effects. This needs to be studied some more.

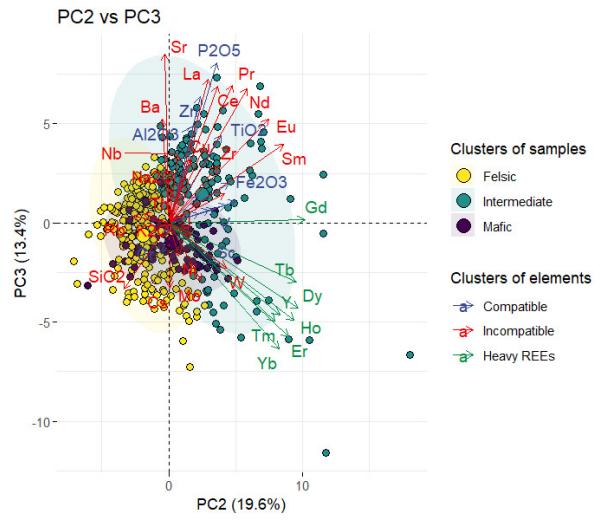


Fig. 9. Clusters according to PC2 and PC3. Large ionic radius (light REEs) and small ionic radius (heavy REEs and compatible elements) are found in element clusters. Felsic, mafic and intermediate elements are found in sample clusters.

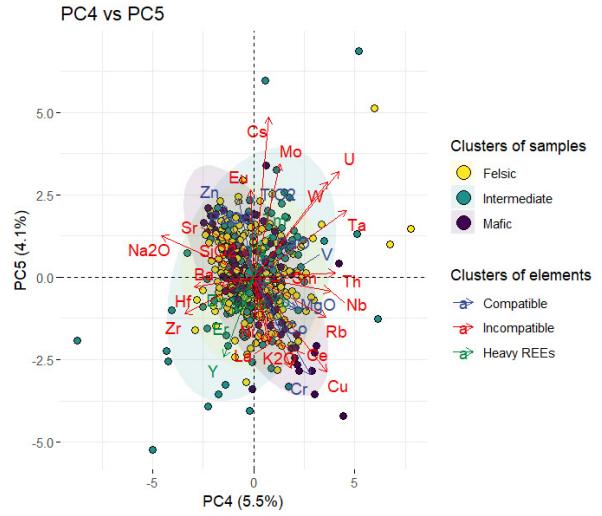


Fig. 10. Clusters according to PC4 and PC5. Mobile alkaline elements and immobile elements are found in element clusters.

#### D. Discussion

For supervised learning, the Decision Trees algorithm obtained the best average metrics results: accuracy: 87%, precision: 89%; recall: 89%; and F-score: 81%. For unsupervised learning, 5 PCs were used to generate the clusters. The plot with PCs 2 and 3 was found to be related to pressure effects, while the plot with PCs 4 and 5 could be related to water effects.

#### IV. CONCLUSIONS AND FUTURE WORK

This research work proposed a methodology to apply machine learning to geochemical data and an open web application for autoML. This tool will allow geoscientists to load geochemical datasets and perform analysis with supervised and unsupervised learning. A dataset composed by granitic rock samples from Southern California was analyzed with both

learning methods using the web application. For supervised learning, the Decision Trees model offered the best results. For unsupervised learning, 2 plots were found that could be related to water and pressure effects. As future work, the proposed pipeline will be extended to apply other techniques to prepare the data, such as downsampling. Also, the web application will incorporate new functions, such as: saving the models created by users to be used more than once, and allowing the tuning of more parameters for supervised and unsupervised learning methods.

## REFERENCES

- [1] J. S. Dramsch, "Chapter one - 70 years of machine learning in geoscience in review," in *Machine Learning in Geosciences*, ser. Advances in Geophysics, B. Moseley and L. Krischer, Eds. Elsevier, 2020, vol. 61, pp. 1 – 55. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0065268720300054>
- [2] D. J. Lary, A. H. Alavi, A. H. Gandomi, and A. L. Walker, "Machine learning in geosciences and remote sensing," *Geoscience Frontiers*, vol. 7, no. 1, pp. 3 – 10, 2016, special Issue: Progress of Machine Learning in Geosciences. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1674987115000821>
- [3] C. M. Sapretti, L. G. da Fonseca, E. Pereira, and L. C. de Oliveira, "Machine learning approaches for petrographic classification of carbonate-siliciclastic rocks using well logs and textural information," *Journal of Applied Geophysics*, vol. 155, pp. 217 – 225, 2018.
- [4] B. Scott and N. C. Steenkamp, (2019, Jul.) Machine learning in geology. [Online]. Available: <https://www.africanmining.co.za/2019/07/29/machine-learning-in-geology/>
- [5] J. MSV. (2018) Why do developers find it hard to learn machine learning? [Online]. Available: <https://www.forbes.com/sites/janakirammsv/2018/01/01/why-do-developers-find-it-hard-to-learn-machine-learning/?sh=d47fe096bf6d>
- [6] S. Vieira, R. Garcia-Dias, and W. Pinaya, *Machine Learning Methods and Applications to Brain Disorders*, 11 2019, ch. A step-by-step tutorial on how to build a machine learning model, pp. 343–370.
- [7] Q. Yao, M. Wang, H. J. Escalante, I. Guyon, Y. Hu, Y. Li, W. Tu, Q. Yang, and Y. Yu, "Taking human out of learning applications: A survey on automated machine learning," *CoRR*, vol. abs/1810.13306, 2018. [Online]. Available: <http://arxiv.org/abs/1810.13306>
- [8] J. Pearce and J. Cann, "Tectonic setting of basic volcanic rocks determined using trace element analyses," *Earth and Planetary Science Letters*, vol. 19, pp. 290–300, 06 1973.
- [9] C. Li, N. T. Arndt, Q. Tang, and E. M. Ripley, "Trace element indiscrimination diagrams," *Lithos*, vol. 232, pp. 76 – 83, 2015. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0024493715002315>
- [10] J. Armstrong-Altrin and S. P. Verma, "Critical evaluation of six tectonic setting discrimination diagrams using geochemical data of neogene sediments from known tectonic settings," *Sedimentary Geology*, vol. 177, no. 1, pp. 115 – 129, 2005. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0037073805000783>
- [11] F. Hutter, L. Kotthoff, and J. Vanschoren, *Automated machine learning: methods, systems, challenges*. Springer Nature, 2019.
- [12] E. Alpaydin, *Introduction to Machine Learning*. MIT press, 2010, ch. What is Machine Learning?, pp. 1–3.
- [13] P. Harrington, *Machine Learning in Action*, 2012, ch. Machine Learning basics, pp. 3–17.
- [14] M. Mohammed, M. Khan, and E. Bashier, *Machine Learning: Algorithms and Applications*, 07 2016.
- [15] A. Goyal. (2019, Aug.) A brief introduction to autoML. [Online]. Available: <https://becominghuman.ai/a-brief-introduction-to-automl-fa6b598d408>
- [16] F. Osisanwo, J. Akinsola, O. Awodele, J. Hinmikaiye, O. Olakanmi, and J. Akinjobi, "Supervised machine learning algorithms: classification and comparison," *International Journal of Computer Trends and Technology (IJCTT)*, vol. 48, no. 3, pp. 128–138, 2017.
- [17] P. Harrington, *Machine Learning in Action*, 2012, ch. Classifying with k-Nearest Neighbors, pp. 18–36.
- [18] ———, *Machine Learning in Action*, 2012, ch. Logistic Regression, pp. 83–100.
- [19] ———, *Machine Learning in Action*, 2012, ch. Splitting datasets one feature at a time: decision trees, pp. 37–60.
- [20] ———, *Machine Learning in Action*, 2012, ch. Support vector machines, pp. 101–128.
- [21] P. Marius, V. Balas, L. Perescu-Popescu, and N. Mastorakis, "Multilayer perceptron and neural networks," *WSEAS Transactions on Circuits and Systems*, vol. 8, 07 2009.
- [22] C. Ding and X. He, "K-means clustering via principal component analysis," *Proceedings. Twenty-First International Conference on Machine Learning, ICML 2004*, vol. 1, 09 2004.
- [23] P. Harrington, *Machine Learning in Action*, 2012, ch. Using principal component analysis to simplify data, pp. 269–279.
- [24] ———, *Machine Learning in Action*, 2012, ch. Grouping unlabeled items using k-means clustering, pp. 207–223.

- [25] K. Itano, K. Ueki, T. Iizuka, and T. Kuwatani, “Geochemical discrimination of monazite source rock based on machine learning techniques and multinomial logistic regression analysis,” *Geosciences*, vol. 10, p. 63, 02 2020.
- [26] K. Maxwell, M. Rajabi, and J. Esterle, “Automated classification of metamorphosed coal from geophysical log data using supervised machine learning techniques,” *International Journal of Coal Geology*, vol. 214, p. 103284, 2019. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0166516219306457>
- [27] D. Hasterok, M. Gard, C. Bishop, and D. Kelsey, “Chemical identification of metamorphic protoliths using machine learning methods,” *Computers & Geosciences*, 08 2019.
- [28] K. Ueki, H. Hino, and T. Kuwatani, “Geochemical discrimination and characteristics of magmatic tectonic settings; a machine learning-based approach,” *Geochemistry, Geophysics, Geosystems*, vol. 19, pp. 1327–1347, 04 2018.
- [29] M. Petrelli and D. Perugini, “Solving petrological problems through machine learning: the study case of tectonic discrimination using geochemical and isotopic data,” *Contributions to Mineralogy and Petrology*, vol. 171, no. 10, pp. 1–15, 2016.
- [30] K. J. Ellefsen and D. B. Smith, “Manual hierarchical clustering of regional geochemical data using a bayesian finite mixture model,” *Applied Geochemistry*, vol. 75, pp. 200 – 210, 2016. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0883292716300920>
- [31] Y. Jiang, H. Guo, Y. Jia, Y. Cao, and C. Hu, “Principal component analysis and hierarchical cluster analyses of arsenic groundwater geochemistry in the hetao basin, inner mongolia,” *Chemie der Erde - Geochemistry*, vol. 75, 02 2015.
- [32] G. H. Alférez, J. Rodriguez, B. Clausen, and L. Pompe, “Interpreting the geochemistry of southern california granitic rocks using machine learning,” 2015.
- [33] P. Geurts, D. Ernst, and L. Wehenkel, “Extremely randomized trees,” *Machine Learning*, vol. 63, pp. 3–42, 04 2006.
- [34] M. Feurer and F. Hutter, *Hyperparameter Optimization*, 05 2019, pp. 3–33.
- [35] M. Grinberg, *Flask Web Development: Developing Web Applications with Python*, 1st ed. O'Reilly Media, Inc., 2014.
- [36] A. K. Baird and A. T. Miesch, “Batholithic rocks of southern california; a model for the petrochemical nature of their source materials,” Tech. Rep., 1984, report.
- [37] R. Hildebrand and J. Whalen, “Arc and slab-failure magmatism in cordilleran batholiths ii à la cretaceous peninsular ranges batholith of southern and baja california,” *Geoscience Canada*, vol. 41, 12 2014.
- [38] P. GROMET and L. T. SILVER, “REE Variations Across the Peninsular Ranges Batholith: Implications for Batholithic Petrogenesis and Crustal Growth in Magmatic Arcs,” *Journal of Petrology*, vol. 28, no. 1, pp. 75–125, 02 1987. [Online]. Available: <https://doi.org/10.1093/petrology/28.1.75>