Chapter 19
Evolutionary Machine Learning in Environmental Science

João E. Batista and Sara Silva

Abstract This chapter reviews the use of Evolutionary Machine Learning (EML) in environmental science. We cover the various steps of the machine learning pipeline, also addressing topics like model robustness, interpretability, and human-competitiveness. Environmental science is an interdisciplinary field mainly dedicated to climate change, natural resource management, conservation biology, and sustainability. We review applications such as forest monitoring, optimization of photovoltaic installations, improvement of traffic flow, and reduction of waste in animal farms, among others.

19.1 Introduction

This chapter covers Evolutionary Machine Learning (EML) methods for data preparation, feature engineering, regression and classification, among other common tasks in environmental science. While going through the different steps of the machine learning pipeline, we also address the robustness, interpretability, and human-competitiveness of the models obtained with EML methods. We follow the definitions proposed in [103], according to which an "interpretable" model can be understood by domain experts without requiring explanation algorithms, thanks to its size, operators, and number of used features. By opposition, some models are not inherently interpretable but can be "explained" using other algorithms.

EML adds flexibility to machine learning algorithms that are already highly versatile. Therefore, it is not surprising that EML is being applied to an endless list of tasks in various scientific domains. In environmental science, however, EML is still underused, with many experts still preferring other methods like random forests and Artificial Neural Networks (ANNs). Nowadays, ANNs are being used in several scientific domains but, although these models provide good results, they are not

J. E. Batista · S. Silva (✉)
LASIGE, Department of Informatics, Faculty of Sciences, University of Lisbon,
1749-016 Lisboa, Portugal
e-mail: sgsilva@fc.ul.pt

W. Banzhaf et al. (eds.), *Handbook of Evolutionary Machine Learning*,
Genetic and Evolutionary Computation,
https://doi.org/10.1007/978-981-99-3814-8_19

interpretable. EML-based models, on the other hand, are generally known for their potential interpretability [11].

Following the description by Miller et al. [83], environmental science is a vast interdisciplinary field that integrates knowledge from several fields such as biology, chemistry, geology, geography, economics, political science, philosophy, and ethics. Its major goal is to understand how nature works, how we interact with the environment, and how to deal with environmental problems and live sustainably. Machine learning is increasingly recognized as a strong ally in tackling environmental issues [101].

In this chapter, we review several examples of EML being successfully applied to different tasks in environmental science. We promote EML by surveying the application of methods based on Genetic Algorithms (GA) [52] for data cleaning; methods based on GA, Ant Colony Optimization (ACO) [57], Genetic Programming (GP) [96], Multi-Objective Evolutionary Algorithms (MOEA) [34], and Particle Swarm Optimization (PSO) [39] for feature selection; methods such as Evolutionary Feature Synthesis (EFS) [7], Feature Engineering Automation Tool (FEAT) [65] and Evolutionary Polynomial Regression (EPR) [46], and methods based on GA and GP, for feature construction; methods based on ACO, PSO, and GA for other tasks, e.g., parameter tuning. We also promote environmental science by surveying an extensive range of applications, among which land cover-type detection, identification of probable forest fire ignition points, and prediction of ground salinity in remote sensing; prediction of solar radiation for photovoltaic installations; improvement of gas turbine performance; animal farm management to avoid waste and other applications. Although not specifically focused on EML, a recent survey [101] also includes several applications of EML in environmental science.

19.2 Data Sources and Types

This field relies on data obtained from two sources: *in situ* measurements and remote sensing. While *in situ* measurements require the sensors to be in contact with the objects under study, remote sensing can obtain measurements without any physical contact. *In situ* measurements can be made using a large variety of instruments that are highly dependent on the study case. As an example, Johari et al. [55] mention that soil characteristics can be obtained using pressure plates, Büchner funnels, tensiometers, pressure membranes, filter paper, and heat dissipation sensors. Remote sensing [30] measurements can also be obtained through different sensors, like sonars or cameras installed in boats, and different types of sensors onboard Unmanned Aerial Vehicles (UAVs) and satellites, including optical and radar sensors. Since *in situ* measurements can be performed using an endless list of instruments that vary from task to task, in Table 19.1, we provide only a list of remote sensing instruments used in our bibliography.

Both UAV and satellite sensors measure radiance in specific wavelength intervals. Each of these intervals is called a band, and each sensor may perform measurements

Table 19.1 List of remote sensing instruments used in the surveyed works

Instrument type	Instruments
Sonars	Simdar EQ60 [110]
UAV Sensors	Compact Airborne Spectrographic Imager (CASI) [29]
	Fabry–Pérot interferometer (FPI) [13]
	Hyperspectral Digital Imagery Collection Experiment (HYDICE) [126]
	Airborne Visible InfraRed Imaging Spectrometer (AVIRIS) [106]
Satellites (SAR[a])	Sentinel-1 [43]; E-SAR, RADARSAT-2 and AIRSAR [71][b]
Satellites (Optical)	Landsat-8 (LS8) [90] and Sentinel-2 (S2) [42]

[a] Synthetic Aperture Radar (SAR)
[b] Described as Polarimetric SAR (PolSAR) imagery

in several different bands. For consistency with the machine learning nomenclature, we will freely call "feature" to each of these bands. UAVs allow the acquisition of very high-resolution imagery (e.g., 10cm per pixel, using the P4 multi-spectral UAV [82]). However, this data is limited in coverage. By opposition, satellite imagery is freely available and offers continuously updated worldwide coverage, for example, with Landsat-8 (LS8) [90] and Sentinel-2 (S2) [42]. The downside of free satellite imagery is the spatial resolution, which tends to be much lower than with UAVs, ranging from 10m to 60 m per pixel on both LS8 and S2.

The data used by the authors in our bibliography can be divided into two categories: uni-temporal and multi-temporal datasets. The first category includes datasets whose measurements were obtained in a single time frame and will be referred to as "uni-temporal datasets", or simply "datasets". The second category includes datasets whose values are obtained using several measurements, obtained in a succession of different time frames, and will be referred to as "time series". Although uni-temporal data usually provides good results, time series normally contains more useful information and therefore can be used to solve harder tasks. One application mentioned later is the separation of similar types of vegetation through their phenological cycles [16].

This chapter covers the use of EML-based machine learning methods in uni-temporal datasets and time series, in several different tasks. Each section of this chapter covers a different kind of application, in the following order: data preparation, feature engineering (Sect. 19.4), split into feature selection and feature construction; regression tasks (Sect. 19.5), i.e., the prediction of a numerical value for each sample, like the amount of biomass in each location; classification tasks (Sect. 19.6), split into binary and multiclass classification, i.e., prediction of the nominal label of each sample, like the land cover type for each pixel in satellite imagery; other tasks (Sect. 19.7) like image alignment and parameter tuning. Other examples of applications are then given, focusing on the robustness of the EML models (Sect. 19.8) and highlighting the human-competitiveness of results obtained in selected work by the authors (Sect. 19.9).

19.3 Data Preparation

The machine learning pipeline includes several steps that should be followed to ensure the quality of the final model. After collecting the data, the first step is data preparation. Data preparation includes tasks such as dealing with missing values, removing duplicated samples and outliers, and data normalization, among other tasks that may depend on the algorithms to be used later on, e.g., one-hot encoding to convert categorical features into numerical or boolean features.

Dealing with missing values is normally done using statistical methods. However, some authors use machine learning, and a recent survey [49] provides a list of machine learning techniques used for this task. From the EML field, only methods based on GAs are mentioned, namely, the Multi-Objective GA for data Imputation (MOGAImp) [73]. Two other works using EML for missing value imputation describe the use of GAs [44] and ANNs assisted with GAs to minimize their error function [2]. Specific to environmental science, we only find two EML works that perform missing value imputation: one uses GAs for the measurement of gas turbine blades [40] and the other is the MOGAImp method [73] applied to the classification of land cover types using satellite imagery.

In some cases, the presence of a few bad samples within a dataset can lead to overfitting. Although their work is not focused on data preprocessing, in [111] the authors propose a semi-supervised GP method that effectively detects and avoids learning mislabeled data in a dataset regarding the detection of burnt areas in satellite imagery. Such methods can also be studied from the data preparation point of view.

19.4 Feature Engineering in Environmental Science

After data preparation, feature engineering is the next step in the machine learning pipeline. Feature engineering is necessary to ensure the machine learning algorithms use high-quality data to induce their predictive models, leading to better results and less overfitting [35]. This step usually requires an expert to indicate which variables correlate with the problem, which variables we should discard and which variables we should combine (e.g., to create indices that highlight characteristics of the dataset samples). This is traditionally a manual task that requires domain-specific knowledge, therefore it is time-consuming. However, feature engineering is becoming more automated, namely, with its integration in AutoML systems like TPOT [91]. According to [72], feature engineering can be partitioned into feature selection and feature construction.[1] This section addresses both types.

[1] Frequently also called feature extraction, feature generation, feature learning, feature discovery, feature synthesis, or constructive induction.

19.4.1 Feature Selection

Feature selection deals with reducing the number of features within a dataset. The objective is to facilitate the learning of the machine learning algorithm or to reduce the likelihood of overfitting by removing redundant or noisy features from the dataset [72]. Although this selection is usually made by experts using their domain-specific knowledge, many methods are being developed to perform this task automatically.

Over the years, many EML algorithms have been used to perform feature selection. GAs [52] are probably the oldest subtype of EML to be used for feature selection [68] and continue to be a very popular choice nowadays. There are other popular EML methods for feature selection, such as the ACO [57], GP [96], MOEA, and PSO [39] algorithms.

As a practical example of feature selection, consider a GA in a hypothetical dataset with 10 features, where eight are noise and two are necessary to reach a good solution. Following the illustration in Fig. 19.1, the GA randomly generates a population of possible solutions to the problem in the form of arrays of "allowed features" to be used by a decision tree and uses their accuracy on a validation set as fitness. Over the generations, the noisy features are dropped out of the population while the good features survive. In a perfect scenario, the final model uses all the good features and none of the noisy ones.

We now focus on methods created and studied to perform feature selection in environmental science tasks. In remote sensing aerial imagery, feature selection is commonly applied to select relevant bands in an image out of dozens or hundreds of bands. In [29], the authors use the CASI imager to generate high-resolution images using 72 spectral bands from 409 to 947nm. Then, the authors compare the feature selection performed by GP-SVI (their proposed GP-based method) and GA-PLS (a GA-based method) with traditional statistical methods, concluding that GP-SVI outperforms the traditional methods and that GA-PLS, although not outperforming the traditional methods, has a very low standard deviation on the test results. Similarly, in [51], the authors study three datasets obtained from images with a large number of

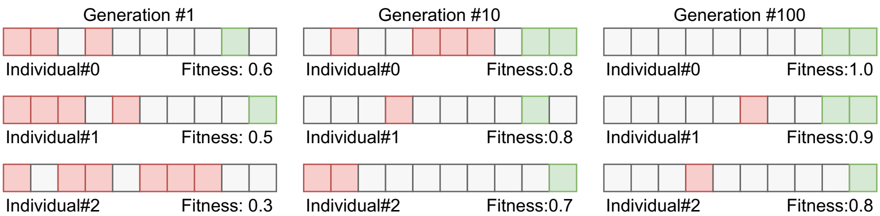


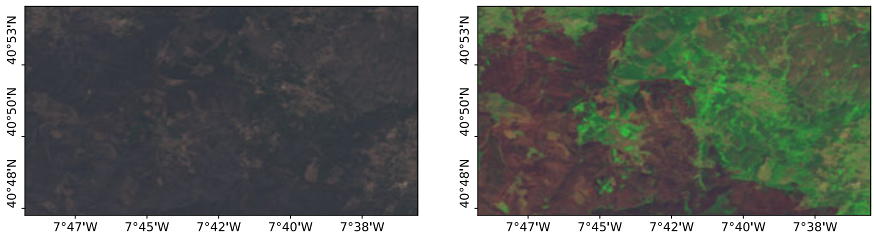
Fig. 19.1 Example of feature selection using a GA. Using a fitness function that promotes selecting the good (green) features while avoiding the noisy (red) features, the GA learns which features are useful for the problem over the generations. A possible fitness function is the performance of a learning algorithm using only the selected features, e.g., validation accuracy of a decision tree

bands (115, 220, and 224 bands). They compare HC-ABC (their proposed method, Hypergraph Clustering Artificial Bee Colony) with four non-EML algorithms and two EML algorithms and conclude that their method outperforms the other six in all three datasets. The PSO algorithm is used in [13] to select 15 bands out of the 380 bands available using the FPI sensor, and in [126] to select 7 bands out of 180 using the HYDICE sensor. Lastly, in [106], the authors compare 10 different EML algorithms for feature selection in three images obtained from the AVIRIS sensor with 176, 200, and 204 bands.

Overall, this illustrates the potential of EML-based algorithms for feature selection in datasets with a large number of features. While these works are specific for datasets obtained using UAV imagery, datasets with many more features exist, and works from other areas show that EML-based algorithms can find optimal features in datasets with numbers of features that range from one thousand to over one million [22, 66, 94, 99, 118, 122].

Unlike UAV imagery, satellite imagery from the most popular satellites has a smaller number of bands. For example, the LS8 and S2 satellites only use 11 and 13 bands, respectively. Since satellites usually use a smaller number of bands, the gap between the wavelengths measured by each band tends to be bigger. This results in satellite bands having well-known and different meanings, as explained in detail in [90]. As described in this web page about the LS8 satellite, healthy vegetation is very reflective in Near-Infrared (NIR) wavelengths (band 5), so this band is very important to study vegetation. Shortwave Infrared (SWIR, bands 6 and 7) bands are very useful to study the soil, and band 7 allows an easier detection of burnt areas, as illustrated in Fig. 19.2. Although satellite imagery datasets usually have a small number of features, it is useful to have feature selection methods that remove unnecessary bands from the dataset since this usually leads to the creation of simpler machine learning models.

In [56], the authors build an improved version of the Normalized Difference Vegetation Index (NDVI) [102] using a GA to evolve weights in a mathematical expression that uses several bands. After obtaining the final weights, the authors removed the bands that had little impact on the expression (i.e., the bands associ-



(a) RGB image (b) RGB image obtained using bands 7, 5 & 1

Fig. 19.2 Landsat-8 satellite imagery over Guarda (Portugal) in September 2022. The image in (*b*) uses band 7 to highlight burnt areas (brown) and band 5 to highlight healthy vegetation (green), facilitating the detection of burnt areas ated with very small weights). In [71], the authors propose SAE-MOEA/D, a new MOEA algorithm with a Stacked AutoEncoder (SAE), to select bands to be used by convolutional neural networks (CNN) [47] in PolSAR imagery.

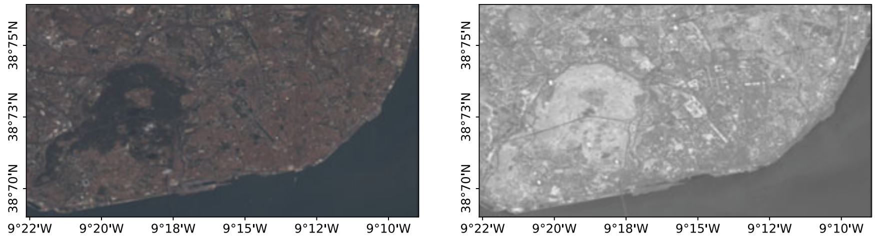
Some methods solve tasks by evolving models that do not use all the available features. Since feature selection is a collateral result of their primary objective, we consider this "implicit" feature selection. Since some GP-based algorithms evolved models that only use a subset of the features, this implicit feature selection is a frequent side effect of their evolution. The following works perform implicit feature selection, although their main objective is feature construction. As such, we will explain them in greater detail in Sect. 19.4.2.

In [16, 18, 82], the authors use GP-based methods and comment on the features selected for the final models. In [82], the authors report that the red and blue wavelengths seem to be highly preferred by the models when detecting mistletoe, while the red-edge infrared and NIR wavelengths seem to be avoided. In [16], the authors use time series to separate two similar land cover types using data from several months and comment that the features of August are frequently selected by the models, indicating that this month is important in the separation of the two land cover types. In [18], the authors evolve features that detect water in cloudy imagery and comment that, besides the models selecting the same features as remote sensing experts do to detect water, they also use the features that are associated with detecting clouds. Other works also use GP-based algorithms for classification [15, 17, 76] and regression [55, 93, 97, 115] tasks. However, the authors do not give emphasis on the feature selection side of these algorithms.

These algorithms can also be applied to feature selection in datasets that are not based on remote sensing. In [127], the authors apply the Standard GP (Std-GP) [96] algorithm to a regression dataset to predict fish weights based on the measurements of the fish. Then, the authors comment on the use frequency of each feature for each fish species, indicating that the weight of each fish species is correlated with the measurements of different body parts. In [38], the authors propose the evolutionary feature selection algorithm and compare this method's accuracy and number of selected features with other EML-based algorithms in a benchmark that, among other non-EML datasets, include a dataset for the classification of animal types based on their physical characteristics. Although this work is not focused on conservation biology, this dataset may help to study habitats in future.

19.4.2 Feature Construction

Feature construction combines or modifies features to create more informative ones [72]. Ideally, these new features are easier to handle by machine learning algorithms and, assuming their interpretability, also by human experts. However, due to the large number of possible combinations of features and the computational cost of testing all combinations, this task is usually performed manually by experts. In this section, we talk about EML for automatic feature construction.



(a) RGB image (b) NDVI image

Fig. 19.3 Landsat-8 satellite imagery, obtained over Lisbon (Portugal) in September 2022. The NDVI in (*b*) is used to highlight the vegetation within the image

Domain experts can use their knowledge to perform manual feature construction, selecting features that are considered relevant to the problem and combining them. For example, in the remote sensing field, experts know that some ratios between certain wavelengths are informative for specific problems. Those ratios can be added to the datasets, thus facilitating learning. The NDVI [102], displayed in Eq. 19.1, is an example of a popular ratio. This popular index highlights healthy vegetation in remote sensing data, as exemplified in Fig. 19.3.

$$NDVI = \frac{NIR - Red}{NIR + Red} \tag{19.1}$$

Although feature construction is traditionally a manual task, over the years, many evolutionary computation methods have been developed to perform automatic feature construction (e.g., EFS [7] and FEAT [65]). Depending on the feature construction algorithm and the complexity of the problem, the evolved set of features varies in the number of features, their complexity, and the number of original features used in each new feature's expression. This leads to some variance in the interpretability of the final models. Some machine learning algorithms, namely those based on deep learning [47], are known not only for their state-of-the-art results but also for their lack of model interpretability, despite some visualization methods [128] and other recent efforts [120, 129] that allow them to be explained. By opposition, EML algorithms are known for their potential interpretability [11, 103]. It should be noted that some EML algorithms, like the ones of neuroevolution, evolve models that are not considered interpretable. Some works may also use algorithms known for providing interpretable models and fail to obtain them due to, e.g., the complexity of the problems or the use of sub-optimal parameters.

As a practical example of feature construction, consider the Std-GP algorithm in a hypothetical dataset where the optimal solution is the sum of two specific features and the other features are noise. Following the illustration in Fig. 19.4, the Std-GP randomly generates a population of possible solutions in the form of mathematical expressions that convert a dataset sample into a single numerical value. Over the generations, the features selected by the model (implicit feature selection) and the

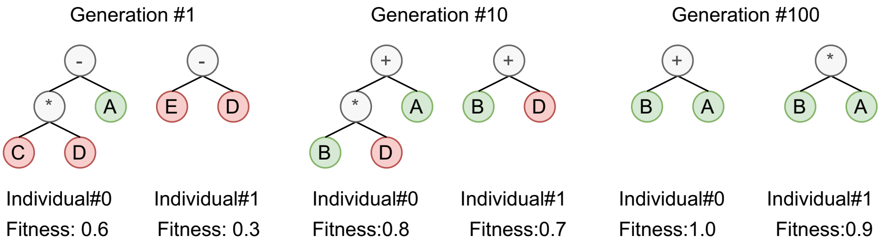


Fig. 19.4 Example of feature construction using Std-GP. Here, the optimal solution is A+B. Using a fitness function that promotes the search for this expression (e.g., accuracy), the population evolves models that select the correct features (implicit feature selection) while using the correct operators

operators and shape of the model (feature construction) will change while optimizing the fitness function. In this case, the fitness function can be the correlation between the model's output and the prediction, making a model applicable to regression tasks and binary classification (e.g., by converting class labels to numerical values 0 and 1). More complex methods, like M3GP [85], evolve a set of trees, resulting in multidimensional model output. Similarly to the GA in the previous section's example, these models may use a second learning algorithm to calculate their fitness. Due to the multidimensionality of the output, these models can also be used for feature construction in multiclass datasets.

The typical way to perform feature construction involves the creation of features and adding them to the dataset, as seen in several works that add manually created features to the dataset [15, 95]. Although some works replace the original features with those evolved by the machine learning algorithms [16, 17, 75], other works suggest that replacing the original features might not always be the best option. Instead, these works add the evolved features to the dataset [15]. However, depending on the classifier, this approach should be avoided since it increases the dimensionality of the dataset, possibly leading to a higher computational cost when training other models in the extended dataset.

The following works dealt with the evolution of features whose expression is small enough to be considered easily interpretable. In [31], the authors use a GA to evolve weights for an arithmetic expression that combines three features and three weights, with the objective of mimicking the NDVI using the RGB values. This allows the use of cameras without infrared sensors in the study of healthy vegetation. Similarly, in [56], the authors create an improved version of the NDVI by evolving a set of 12 weights and, after training the model, remove the features associated with small weights, simplifying the expression evolved by the algorithm.

The previous works use GA-based algorithms to evolve features with a fixed structure predefined by authors. In opposition, GP-based algorithms do not require a predefined structure for the models since they are mutable and adapted over the evolutionary cycle. Even among GP-based algorithms, the structure of the models can be decomposed into two types: models composed of a single tree and models composed of multiple trees. While the first approach is usually far more limited, it can be applied to regression and binary classification tasks in a straightforward way, while multiclass classification tasks usually require multiple features, i.e., a model with multiple trees/outputs.

Typically, the authors use GP-based algorithms, namely the Std-GP algorithm, to evolve models composed of a single tree, both for regression tasks [55, 93, 97, 113, 115, 127] and for binary classification [18, 21, 76, 123]. Other applications of GP-based methods in remote sensing can also be seen in [67]. Unlike Std-GP, other EML-based feature construction methods, such as the already mentioned EFS [7], EPR [46] and M3GP [85] algorithms, evolve a set of trees, rather than a single tree, resulting in a multidimensional output from the model. Although these algorithms show potential in creating interpretable models, this interpretability may depend on the number of trees and the mathematical operators used. In [81], the authors review several works related to feature construction using GP-based methods in the context of explainable artificial intelligence. Those works have a wide range of application fields, including environmental science.

For more complex tasks, such as multiclass classification and harder binary classification and regression tasks, the authors typically use methods that evolve several features. These features can work independently from each other or in a cooperative manner. In works that treat multiclass classification as a succession of binary classification problems [110], we consider that the features are independent, since each feature separates a different set of classes. In more advanced multiclass classification methods, such as the M3GP [85] and M4GP [66], and the Std-GP algorithm with COMB functions [75], the algorithms evolve a set of cooperative features (a transformation) that converts the feature space, and then perform multiclass classification in this new space, separating all classes using the same transformation. Another method derived from M3GP, called eM3GP [109], assumes that an ensemble of transformations, each proving to be good for discriminating a single class, may do a better job than a single transformation used for discriminating all the classes. In [87], a set of features is evolved to be jointly used in a single linear regression model.

19.5 Regression Tasks for Environmental Science

In regression tasks, the machine learning methods attempt to predict a numerical value to be associated with each sample in a dataset. Taking the NDVI [102] as an example, this index associates each pixel in an image with a value from -1 to 1, where a higher value indicates healthier green vegetation. In a regression task to detect vegetation, a machine learning model should have a similar kind of behavior. In this section, we will give an overview of works related to the detection of vegetation and salinity levels, the prediction of necessary soil nutrients for crop development, the prediction of soil properties, and the prediction of solar radiation.

The most common application of regression models in environmental science is precisely the creation of models to detect some property on the terrain, such as vegetation, biomass, or salinity. In [97, 117], the authors apply the Std-GP algorithm to satellite imagery to create vegetation indices that correlate with the Revised Universal Soil Loss Equation (RUSLE) C factor, an index that assesses how land use affects soil loss and sediment generation [8]. The authors improve previous work that could only be applied to green vegetation by creating models that also correlate with the RUSLE C factor in dry vegetation. In [31], the authors use GAs to evolve weights to be used in a function that tries to predict the NDVI values using only the visible wavelengths sensor (i.e., without using the NIR wavelength). Similarly, in [56], the authors use GAs to evolve weights to create a new vegetation index. In [5], the authors use the Std-GP algorithm to evolve indices for phenology analysis using time series data obtained from a hemispherical lens camera set in an 18-meter tall tower. In [28], Std-GP is used to predict levels of chlorophyll concentration, obtaining better results than traditional methods. In [108], the authors apply 14 machine learning methods, including Std-GP and Geometric Semantic GP (GSGP) [84] in the prediction of forest biomass. With this work, the authors conclude that even the best methods overfit the training data, indicating that this is a complex task, especially when using less than 100 samples for training. In [3], the authors use a GA-based algorithm to recommend nutrients for optimal crop development, improving soil fertility using time series, resulting in increased production. Similarly, in [29], the authors use a GP-based algorithm for precision farming, obtaining better results than those obtained using traditional approaches. In [130], the authors use GA to optimize the number of hidden layers and nodes in a CNN. In this work, the model is applied to the prediction of evapotranspiration values of soybean plantations in different growth stages. In [1], the authors use Std-GP to predict the fraction of calves that survive per cow each year in several farms. This work on precision farming leads to a better estimate of the number of calves that the farms should produce to minimize waste and maximize profits.

Some works are related to the application of the Std-GP algorithm to the prediction of soil properties. In [115], the authors use satellite imagery to predict the soil electrical conductivity and soil surface salinity. In [55], the authors study the Soil–Water Characteristic Curve (SWCC) of different kinds of soils from several soil properties. In [93], the authors study the saturated hydraulic conductivity of different soil types.

As an important theme in environmental science, several works focus on predicting several aspects related to energy. In [45], the authors use several EML algorithms to estimate the amount of solar radiation that would hit Queensland (Australia) using time series obtained from monthly predictors over 5 years. Similarly, in [9], the authors study the prediction of solar power for the integration of a photovoltaic site to improve the reliability of photovoltaic systems using time series, with measures of solar power every 5 minutes, over several days, in three different cities in Florida (USA). In [89], the authors use GAs to search for the optimal locations for photovoltaic installations in La Palma Del Condado (Spain) under a series of constraints to minimize environmental, safety, and economic risks. Those constraints make the algorithm avoid suggestions near environmentally protected areas, roads, urban areas, and vineyards (due to their importance to the local economy). Similarly, in [98], the authors also deal with the prediction of solar power production from solar panels, in Potsdam University (Germany). Gas turbines are also of great importance for power generation. However, due to their complexity and their execution environment, they can have a high failure rate that leads to severe consequences. To minimize this problem, in [41], Std-GP and several variants of GSGP are used to predict the fuel flow and the exhaust gas temperature in two separate datasets, to improve the performance of the gas turbines.

Especially in summer, fire is the most destructive force in the Portuguese forest. Typically, there are between 15.000 and 25.000 forest fires each year across the whole country [79]. In [121], the authors use a GA to evolve the weights of ANNs. This model is then used to predict probabilities of fire ignitions in several municipalities. In [26], the authors compare GSGP with other well-known methods, including ANNs, to predict the sizes of burnt areas, while taking into consideration the characteristics of the terrain.

With more than half the world's population living in urban areas, air pollution is another theme of high importance in environmental science. Ozone concentration levels are related to air pollution, and low concentration levels have a negative impact on human health. The authors of [23] propose a new variant to the GSGP algorithm and compare it with GSGP and other well-known methods, including ANNs, to predict ozone concentration levels in the region of Yuen Long, one of the most polluted regions in China. This work proposes a faster method with better results and lower variance across the different runs. In [78], the authors propose a GA-based algorithm based on iterative rule learning. This algorithm is then used to detect atmospheric pollution by evolving a set of rules that are applied to the detection of high ozone concentration levels, nitrogen monoxide and sulfur dioxide. Later, in [77], the authors propose a new GA-based algorithm to evolve association rules for the detection of high ozone concentration levels.

Water-related environmental issues have also been the theme of some works. In [61], the authors use Std-GP to predict the quantity of the *Microcystis aeruginosa* bacteria in the Nakdong River in South Korea. In this work, the authors use two variants of Std-GP. In the first approach, the models exclusively use arithmetical operators, while in the second approach, the models also use the logical IF-THEN-ELSE operators. In [27], the authors use GP models to study the patterns of total phosphorus in Tampa Bay, Florida. In this work, GP models are trained to estimate the changing levels of total phosphorus in the bay and, after validation, are used to generate the map to be studied. In [19], the authors use the EPR algorithm to predict the number of pipe failures in water distribution systems from their attributes.

Some authors used the ENC and ENH datasets [119] to benchmark their methods (MRGP [6], MGP [64], FEAT [65], EPLEX and EPLEX-1M [63] and GSGP [25]). These are regression datasets to predict the energy performance of the residential buildings for cooling (ENC) and heating (ENH).

Finally, in [113], the authors use GP to build interpretable models to predict global mean temperatures. Their results suggest that GP can evolve interpretable models that are comparable to complex climate systems, even without resorting to historical data when making predictions.

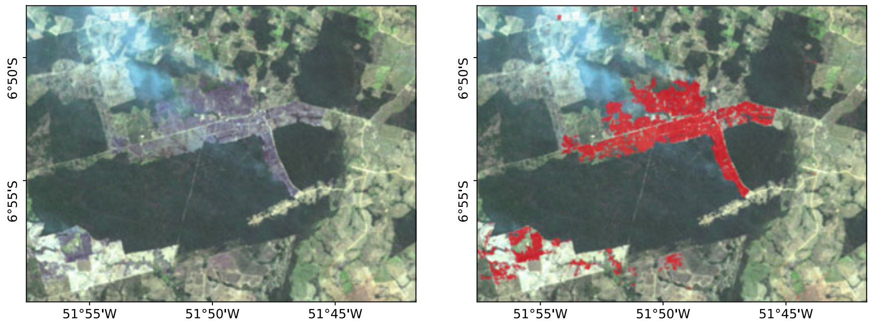
19.6 Classification Tasks in Environmental Science

In classification tasks, the machine learning model has to associate each sample in a dataset with a nominal label. For example, a model trained to detect burnt areas should associate each dataset sample with the label "Burnt" or "Non-burnt". Classification includes two categories: binary classification and multiclass classification. In binary classification, the model has to associate the sample with one of two classes. This task is usually considered simple. Regression models can also perform binary classification by defining a threshold that separates both classes. In multiclass classification, the model has to associate the sample with one of three or more classes.

19.6.1 Binary Classification

Std-GP and other GP algorithms are some of the popular choices for binary classification tasks. These algorithms are used to detect riparian zones using mixed data from two satellites [76], to detect water in cloudy satellite images [18], burnt areas in satellite images [15, 17, 21, 100, 107, 111, 112], to separate forest from savanna [4], and to identify cropland in high-resolution satellite imagery [37, 123]. Binary classification in remote sensing can often be associated with detecting a specific land cover type, as seen in Fig. 19.5, where a GP model is used to detect burnt areas [17]. GP algorithms can also be applied to high-resolution UAV imagery. In [69], the authors apply the Std-GP algorithm to an image segmentation problem to detect algae in river images obtained using a sensor onboard a UAV.

Outside the remote sensing area, Std-GP is used for the prediction of production errors in the steel industry [74]. Although this is not a direct application in environmental science, the authors state that a higher success rate in production results in



(a) RGB image (b) Burnt areas, highlighted in red

Fig. 19.5 Landsat-8 satellite imagery, obtained over the state of Pará (Brazil) in August 2015. The red areas marked in **b** highlight the areas classified as "burnt" by the machine learning model lower energy requirements, which leads to fewer greenhouse gas emissions. Other authors apply GP [70] and GA [124] algorithms to the prediction of traffic congestion.

One of the advantages of using GP-based methods to detect specific classes is the flexibility of the application of the features produced. In [18], the authors compare the application of water detection indices [88] with those evolved by the Std-GP in a dataset that includes many pixels from cloudy locations. The authors conclude that the evolved features are better than some human-made indices and equally good to others. From a practical point of view, the features evolved by [76] can be considered indices to detect riparian zones. This statement implies that GP can produce indices that detect highly difficult land cover types, in opposition to detecting water and burnt areas.

Regression algorithms (e.g., Std-GP) usually produce a model equivalent to a mathematical expression. Since their output is numerical, it requires a final post-processing to perform binary classification. The typical approach is the use of a threshold that separates the two classes, as seen in [18, 100]. Since the classes are nominal and usually do not have an order, this approach is often not applicable to multiclass classification. An exception to this rule may be the discrimination between classes that are related to each other, e.g., barren areas, grasslands, and forests. In this example, we know that the classes have different biomass volumes, which can be used to easily order and classify the samples by using multiple thresholds [50].

19.6.2 Multiclass Classification

Multiclass classification is considered a more complex task, and two main approaches exist to solve it. The more direct option is to use a classifier that can perform multi-class classification by separating all classes at once, using a single model. However, these models are usually complex. Experts may decide to decompose a multiclass problem into several simpler, binary classification problems. There are multiple ways of doing this. The most common approach is the one-versus-all [110] strategy. This approach uses a divide-and-conquer strategy with an *n*-class problem divided into several binary classification problems. As exemplified in Fig. 19.6, the one-versus-all strategy performs a succession of binary classifications until the correct class is identified.

Some works apply this divide-and-conquer strategy to solve classification problems. In [110], the authors study a dataset obtained from sonar data to identify five types of seafloor habitats. By applying the one-versus-all strategy, the authors use four models to first separate sand from other habitats, then reef, then algae, and lastly to separate the *australis* and *sinuosa* seagrasses. In [32], the authors separate several species of vines by applying several methods, including Std-GP, using time series data that measure the intensity of fluorescence emissions using the Kautsky effect [58].

Many recent works focus on the application of multiclass classification models, leading to a simpler pipeline. In [24], the authors apply the Std-GP and GSGP

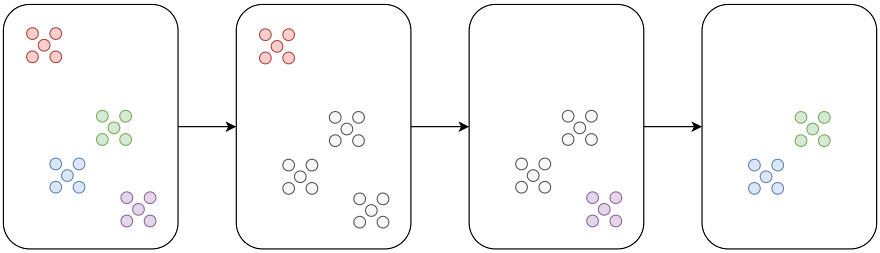


Fig. 19.6 One-versus-all strategy for multiclass classification. In this example, the prediction is made using, at most, three classifiers

algorithms for several multiclass datasets, including two datasets to detect land cover types. Although the GSGP produces better scores, both methods have low accuracy in a dataset containing land cover types in agroforest mosaics, reinforcing the difficulty of identifying plantations in agroforest land cover types. In [71], the authors propose the SAE-MOEA/D algorithm and use it to identify from three to five classes in several radar images from different sensors. In [12], the authors use a multi-objective GA algorithm (NSGA-II [33]) to identify croplands in satellite imagery. In [75], the authors propose a novel GP-based approach to evolve multiple features, testing it on a multiclass time series dataset. The evolved features are tested with several classifiers and the results indicate that these features are more robust than the traditional remote sensing indices developed by experts. In [17], the authors use the M3GP algorithm to detect burnt forest areas in satellite images. This algorithm created models that reduce the dimensionality of the problem from 10 features to 3 features. Besides increasing the accuracy of the machine learning models, this allowed the visualization of the new dataset in a 3D space. In [15], the authors use the M3GP algorithm in several remote sensing datasets for binary classification (detection of burnt areas) and multiclass classification (land cover type classification). The multiclass datasets included, among other classes, three subtypes of forest in satellite imagery. While most classes were easily separable, forest classes are often misclassified due to their spectral similarity. In this work, the authors compare the results from feature construction using M3GP with manual feature construction (i.e., using indices) and automatic feature construction using EFS [7] and the non-EML Fast Function Extraction (FFX) [80] algorithm. However, the automatic methods were limited to binary classification datasets. In [16], the authors use the M3GP in a multiclass time series dataset that includes, among other classes, two land cover types with similar spectral signatures, “natural forest” and “cocoa agroforest”. Since cocoa trees are planted in the shade of natural trees, the classes are nearly identical from the (optical) satellite point of view. Based on the analysis of the models and the frequency of selection of each feature, the authors conclude that, although the confusion matrices reveal some confusion between these classes, the features obtained in August are useful to detect cocoa. In [15], the authors use a group of classification datasets to benchmark their approach. The work includes two datasets related to the separation of land cover types that several authors have used as a benchmark to test their algorithms, such as the M2GP [54], the eM3GP [109], the M3GP [85, 86], and the M4GP [66].

In [13], the authors use a PSO-based algorithm to identify six land cover types in a UAV imagery dataset. Other recent works also apply EML-based algorithms to classification in high-resolution aerial imagery with applications such as the identification of multiclass land cover-type detection using an evolutionary neural architecture search algorithm [20] and several EML algorithms [51, 106], and also to the detection of mistletoe using a GP-based approach [82].

19.7 Other Tasks in Environmental Science

Besides applying EML models to regression and classification problems, other authors apply EML methods to image alignment and parameter tuning.

Regarding image alignment, in [125], the authors propose LMACO (Multi-modal continuous ACO with Local search), an ACO-based algorithm that allows multiple images to be stacked automatically. The LMACO method outperforms other EML-based methods such as Comprehensive Learning PSO (CLPSO), Self-adaptive Differential Evolution (SaDE) [92], and the traditional ACO. According to the authors, this method can also be applied to image fusion, object recognition, and change detection.

Another popular application of EML-based models is parameter tuning. The objective of this task is to search for optimal parameters that maximize another algorithm's robustness. In [104], the authors use GA to tune the gain parameters on a Static Synchronous Compensator (STATCOM) to improve the amount of wind energy that is harvestable in their system. In [71], the authors propose a MOEA-based method that automatically finds sets of parameters and hyper-parameters to be used by stacked autoencoders in the classification of land cover types in PolSAR imagery.

In [45], the authors compare several EML and non-EML algorithms when predicting solar radiation. The main algorithms to solve this task are the Extreme Learning Machine (ELM), ANNs, Support Vector Regressors (SVR) [10], and the Multi-Gene GP (MGGP) [36] algorithms. However, except for MGGP, these algorithms rely on other methods for optimization. The authors use three versions of the ELM algorithm: the basic version, the online sequential ELM, and optimization using SaDE. The authors also use two versions of ANNs, where the weights are evolved using either the PSO or the GA. Lastly, the authors use three versions of the SVR, where some hyperparameters are picked using either grid search or PSO or GA. Other authors also apply ANNs to predict solar power [9] while using GA to optimize the model's weights and biases.

In [3], the authors use a GA-based algorithm to find an optimal set of parameters (optimal quantity of several nutrients) to be used for crop production. When compared to the traditional approach, the GA approach improved soil fertility, increasing the production of the crops in the study area.

19.8 Robustness of EML models

One of the known issues in remote sensing data is the radiometric variations across images caused by the presence of shadows, clouds, and moisture on the soil, among others. Consequently, machine learning algorithms may learn to identify a class based on a temporary characteristic or noise. This overfitting to the characteristics of a particular image may result in the model failing to detect the correct land cover types when given a different image.

Some works comment on the robustness of their models by applying them to different satellite images [16, 17] or different sensors [31]. In [31], the authors use GA to evolve the weights of an expression to simulate the NDVI index using visible wavelengths. This expression is applied to data from two quadcopters with different sensors (Matrice 210 and Phantom 4 Pro+), showing that this expression provides good results in multiple sensors with minimal input from the user. In [17], the authors use three burnt area detection datasets that were obtained from three different countries with different characteristics. Each of these datasets has training and test samples. The authors notice that if a model is trained in one dataset, although it has a very high test accuracy in the respective test set, the accuracy is reduced when applying the model to datasets from different locations. In an attempt to mitigate this lack of transferability, the authors show that several classification algorithms are more robust when trained using features evolved by the M3GP algorithm. Another approach used by the authors is to train models using samples from multiple datasets. In [16], the authors test the robustness of their algorithm by validating the models in an area much larger than the one used during the model's training phase. In that work, the lack of transferability is more evident in the non-EML classifiers that, while achieving higher accuracy values on the test set, are surpassed by the M3GP-based approaches when applied to the validation set.

In most environmental science applications, humans are the ones responsible for labeling samples in datasets. Still, datasets are prone to containing mislabeled samples, leading to imperfect datasets. If the machine learning algorithms are unable to identify mislabeled samples, the model will learn the incorrect labels, overfitting this erroneous data. In [111], the authors use a semi-supervised GP-based method that is able to ignore mislabeled samples by making use of unlabeled data together with the labeled set.

19.9 Examples of Human-Competitiveness

Over the years, EML algorithms produced human-competitive results in various applications from different research fields, including environmental science [59]. John R. Koza surveyed a large section of these works in 2010 [62] and, while this survey only contains works until 2009, other works can be seen in the "Humies" Awards website [53]. This award is given in a competition recognizing human-competitive results produced by genetic and evolutionary computation. Besides the winners, the website also displays the work of the other applicants. This includes recent applications in environmental science and the EML methods used, such as the use of NSGA-II and Evolutionary Strategies (ES) in traffic light optimization [105] and for traffic flow optimization [114] using Green Swarm, a method-based on (10+2)-EA [116]. Other applications include predicting soil moisture using interpretable models [14] obtained from differential evolution and GAs, and the use of GP to predict wind damage [48].

As stated in the feature construction Sect. 19.4, machine learning experts spend a long time in feature engineering [35]. Although this task requires domain-specific knowledge, many automatic methods are being produced, simplifying this task. In this section, we follow the work of Batista et al., using their work as an example of human-competitiveness in automatic feature construction for remote sensing data. This is an extension of some of their work on the detection of burnt areas [17], separation of forest subtypes [15], separation of land cover types with nearly identical spectral signatures [16] and detection of water in cloudy satellite images [18].

In [17], the authors use the M3GP algorithm to evolve a set of robust features to detect burnt areas. In some cases, only three features were produced, which allowed the visualization of the new feature space in a 3D space, as shown in Fig. 19.7. These features were then directly applied to satellite images in order to understand what the models are "seeing" when classifying pixels, and it was noticed that some of the evolved features have a particular meaning. When applying the features to a satellite image, the authors noticed that each feature clearly highlights a particular terrain characteristic, rather than producing a noisy map that depends on the other

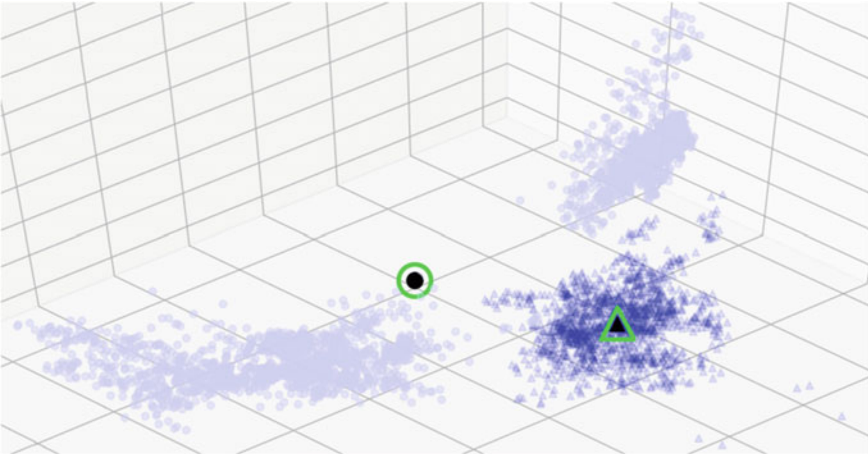


Fig. 19.7 Feature space of a dataset with the "burnt" (dark blue) and "non-burnt" (light blue) land cover types, after being converted by an M3GP model with three dimensions. The centroids of the clusters are marked in black and highlighted in green. This image shows in a clear way that the "non-burnt" cluster should be split into two clusters. *Image source* [17]

features to make sense. In one of the experiments, the final model contained three interpretable features. Two of those features are displayed in Eqs. 19.2 and 19.3. Equation 19.2 seems to highlight burnt areas, areas of low vegetation, and active fire. This statement can be justified by the use of the SWIR2 and Red bands in the equation. Equation 19.3 seems to highlight burnt areas, active fires, and water. This statement can also be justified by the bands selected by the M3GP algorithm for these features. By multiplying these two features, the resulting expression visibly highlights the burnt area better than the Normalized Burn Ratio (NBR) [60], an index of burn developed and used by remote sensing experts. This comparison is seen in Fig. 19.8.

$$(Blue + Red)/SWIR2 \tag{19.2}$$

$$(SWIR2 * Blue)/(Red * NIR) \tag{19.3}$$

In this case, the algorithm evolves features to detect burnt areas and, in [18], to detect water. Remote sensing experts have studied both tasks and indices to detect burnt [60] and water [88] exist. In [18], Std-GP is used to evolve a single feature that detects water pixels while being robust to cloud pixels. The authors study one dataset obtained from two satellite images over the Amazon River's delta and over Portugal. The accuracy values from five water indices are calculated and compared with the ones obtained with Std-GP. The results reveal that the evolved features are statistically better than three of the five human-made indices, and no statistically significant difference was found when comparing the results with the other two indices. Interestingly, Std-GP tends to pick the same features as those used by remote sensing experts, suggesting it can generate similar features without domain-specific knowledge.

In [15, 16], the M3GP algorithm evolves features that discriminate several similar land cover types (e.g., different forest types), a task that is harder than detecting burnt or water, or even simply detecting a forest class without subtypes. In the first article [15], several machine learning methods are used for automatic feature construction. The EFS, FFX, and M3GP algorithms are applied for binary classification (detection of burnt areas), and only the M3GP is applied for multiclass classification (mapping of land cover types). The results obtained using these features are compared with those obtained using indices created by remote sensing experts and without using feature construction methods. Given the difficulty of this multiclass classification task, and the fact that the M3GP-evolved features improved the generalization of the state-of-the-art classifiers, it seems clear that this kind of algorithm can compete with experts and their domain-specific knowledge in the development of indices.

In the second article [16], the authors compare the results from several classifiers (M3GP, multi-layer perceptron, random forest, ridge, ROCKET, and XGBoost) in three versions of a dataset containing several land cover types. One version is a uni-temporal dataset and two other versions are time-series datasets containing data from May to September. The data includes two land cover types with nearly

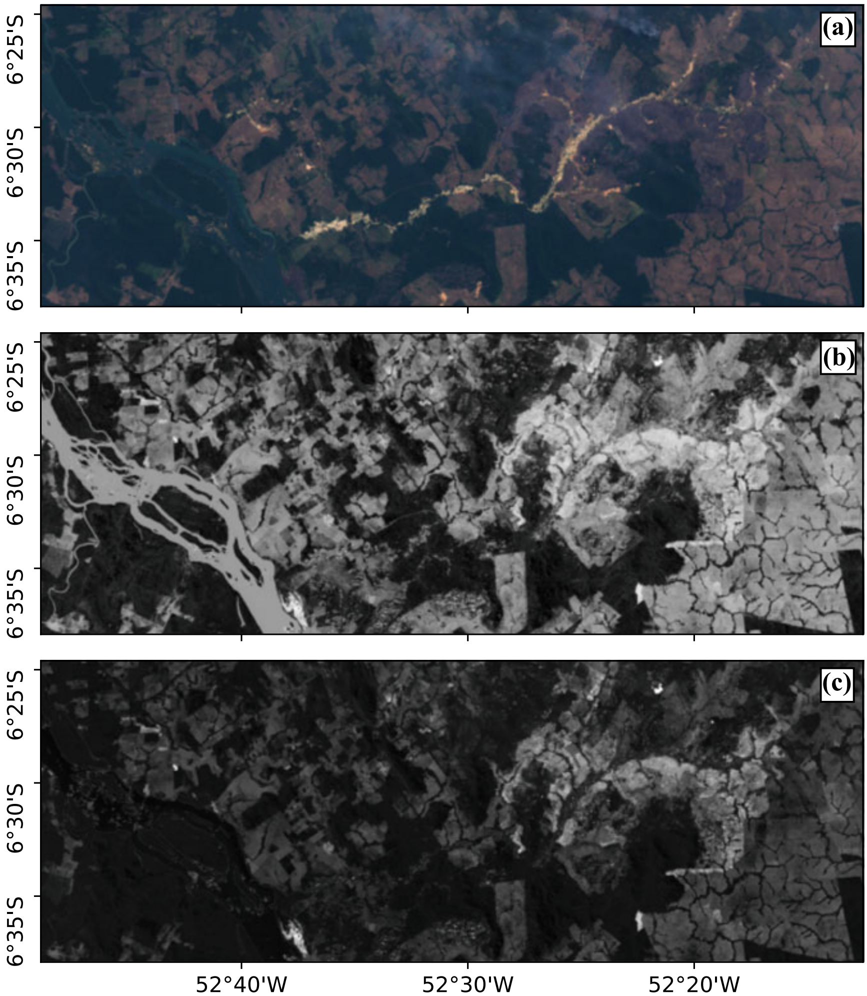


Fig. 19.8 Landsat-8 satellite imagery over Pará, Brazil during the 2015 wildfires. Figure **a** shows the RGB image; **b** shows the application of the NBR index to highlight burnt areas; **c** shows the application of the multiplication of Eqs. 19.2 and 19.3. Both (**b**) and (**c**) were normalized using their maximum and minimum values of the dataset used by the authors in [17]

identical spectral signatures: "forest" and "cocoa-agroforest". Based on the positive results of using M3GP-evolved features in the previous work, the authors study the features generated by M3GP for this new task and learn that, although they do not significantly improve the results of the best classifier, M3GP frequently selects the features from August in the time series data. This reveals that this particular month is relevant to solve the problem. It also highlights the feature selection capabilities of the M3GP algorithm.

19.10 Conclusions

This chapter reviewed the use of Evolutionary Machine Learning (EML) in environmental science applications. The type of data used in these applications varies in several factors, such as source (*in situ* or remote sensing), temporality (uni-temporal or time series) and dimensionality (low to high number of features), and the applications include several different tasks (regression, classification, data cleaning, feature engineering, hyper-parameter optimization, image alignment, among others). This variety of data and application types shows the flexibility of EML.

From the papers reviewed, we observe that genetic algorithms and genetic programming have been popular choices in specific fields of environmental science. For at least three decades, genetic algorithms have been used for automatic feature selection, producing great results in high-dimensional datasets. Algorithms based on genetic programming are also popular choices for automatic feature construction applications. Genetic programming algorithms can be considered human-competitive since they are applied to feature construction tasks where they outperform the experts in the field. Among the EML-based algorithms, they also seem to be a popular choice for classification tasks.

In conclusion, even though EML is still not the first choice, or regarded as state of the art, for environmental science applications, many EML-based methods have been created and used in environmental science applications over the last decades. This reveals a fairly large interest from experts in this area, most probably connected to the advantageous characteristics of EML such as robustness and interpretability.

Acknowledgements This work was supported by the FCT, Portugal, through funding of the LASIGE Research Unit (UIDB/00408/2020 and UIDP/00408/2020); João Batista was supported by PhD grant SFRH/BD/143972/2019.

References

1. Abbona, F., Vanneschi, L., Bona, M., Giacobini, M.: A GP approach for precision farming. In: 2020 IEEE Congress on Evolutionary Computation (CEC). IEEE, July 2020
2. Abdella, M., Marwala, T.: The use of genetic algorithms and neural networks to approximate missing data in database. In: IEEE 3rd International Conference on Computational Cybernetics, 2005. ICCC 2005, pp. 207–212 (2005)
3. Ahmed, U., Chun-Wei Lin, J., Srivastava, G., Djenouri, Y.: A nutrient recommendation system for soil fertilization based on evolutionary computation. Comput. Electron. Agric. **189**, 106407 (2021)
4. Almeida, A.E., da Torres, R.S.: Remote sensing image classification using genetic-programming-based time series similarity functions. IEEE Geosci. Remote Sens. Lett. **14**(9), 1499–1503 (2017)
5. Almeida, J., dos Santos, J.A., Miranda, W.O., Alberton, B., Morellato, L.P.C., da Torres, R.S.: Deriving vegetation indices for phenology analysis using genetic programming. Ecol. Inf. **26**, 61–69 (2015)

6. Arnaldo, I., Krawiec, K., O'Reilly, U.-M.: Multiple regression genetic programming. In: Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation. ACM, July 2014
7. Arnaldo, I., O'Reilly, U.-M., Veeramachaneni, K.: Building predictive models via feature synthesis. In: Proceedings of the 2015 Annual Conference on Genetic and Evolutionary Computation, pp. 983–990, July 2015
8. Asadi, H., Dastorani, M.T., Khosravi, K., Sidle, R.C.: Applying the C-Factor of the RUSLE model to improve the prediction of suspended sediment concentration using smart data-driven models. Water **14**(19) (2022)
9. Asrari, A., Wu, T.X., Ramos, B.: A hybrid algorithm for short-term solar power prediction—sunshine state case study. IEEE Trans. Sust. Energy **8**(2), 582–591 (2017)
10. Awad, M., Khanna, R.: Support vector regression. In: Efficient Learning Machines, pp. 67–80. Apress (2015)
11. Bacardit, J., Brownlee, A.E.I., Cagnoni, S., Iacca, G., McCall, J., Walker, D.: The intersection of evolutionary computation and explainable AI. In: Proceedings of the Genetic and Evolutionary Computation Conference Companion, GECCO '22, pp. 1757–1762, New York, NY, USA. Association for Computing Machinery (2022)
12. Bandyopadhyay, S., Maulik, U., Mukhopadhyay, A.: Multiobjective genetic clustering for pixel classification in remote sensing imagery. IEEE Trans. Geosci. Remote Sens. **45**(5), 1506–1511 (2007)
13. Banerjee, B.P., Raval, S.: A particle swarm optimization based approach to pre-tune programmable hyperspectral sensors. Remote Sens. **13**(16), 3295 (2021)
14. Basak, A., Mengshoel, O.J., Schmidt, K., Kulkarni, C.: Wetting and drying of soil: From data to understandable models for prediction. In: 2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA). IEEE, Oct. 2018
15. Batista, J.E., Cabral, A.I.R., Vasconcelos, M.J.P., Vanneschi, L., Silva, S.: Improving land cover classification using genetic programming for feature construction. Remote Sens. **13**(9), 1623 (2021)
16. Batista, J.E., Rodrigues, N.M., Cabral, A.I.R., Vasconcelos, M.J.P., Venturieri, A., Silva, L.G.T., Silva, S.: Optical time series for the separation of land cover types with similar spectral signatures: cocoa agroforest and forest. Int. J. Remote Sens. **43**(9), 3298–3319 (2022)
17. Batista, J.E., Silva, S.: Improving the detection of burnt areas in remote sensing using hyper-features evolved by M3GP. In: 2020 IEEE Congress on Evolutionary Computation (CEC). IEEE, July 2020
18. Batista, J.E., Silva, S.: Evolving a cloud-robust water index with genetic programming. In: Proceedings of the Genetic and Evolutionary Computation Conference Companion. ACM, July 2022
19. Berardi, L., Giustolisi, O., Kapelan, Z., Savic, D.A.: Development of pipe deterioration models for water distribution systems using EPR. J. Hydroinf. **10**(2), 113–126 (2008)
20. Broni-Bediako, C., Murata, Y., Mormille, L.H., Atsumi, M.: Evolutionary NAS for aerial image segmentation with gene expression programming of cellular encoding. Neural Comput. Appl. **34**(17), 14185–14204 (2021)
21. Cabral, A.I.R., Silva, S., Silva, P.C., Vanneschi, L., Vasconcelos, M.J.: Burned area estimations derived from landsat ETM+ and OLI data: comparing genetic programming with maximum likelihood and classification and regression trees. ISPRS J. Photogramm. Remote Sens. **142**, 94–105 (2018)
22. Carvalho, P., Ribeiro, B., Rodrigues, N.M., Batista, J.E., Vanneschi, L., Silva, S.: Feature selection on epistatic problems using genetic algorithms with nested classifiers. In: Applications of Evolutionary Computation, pp. 656–671. Springer Nature Switzerland (2023)
23. Castelli, M., Gonçalves, I., Trujillo, L., Popovič, A.: An evolutionary system for ozone concentration forecasting. Inf. Syst. Front. **19**(5), 1123–1132 (2016)
24. Castelli, M., Silva, S., Vanneschi, L., Cabral, A., Vasconcelos, M.J., Catarino, L., Carreiras, J.M.B.: Land cover/land use multiclass classification using GP with geometric semantic operators. In: Applications of Evolutionary Computation, pp. 334–343. Springer, Berlin, Heidelberg (2013)

25. Castelli, M., Trujillo, L., Vanneschi, L., Popovič, A.: Prediction of energy performance of residential buildings: a genetic programming approach. Energy Build. **102**, 67–74 (2015)
26. Castelli, M., Vanneschi, L., Popovič, A.: Predicting burned areas of forest fires: an artificial intelligence approach. Fire Ecol. **11**(1), 106–118 (2015)
27. Chang, N.-B., Xuan, Z., Yang, Y.J.: Exploring spatiotemporal patterns of phosphorus concentrations in a coastal bay with MODIS images and machine learning models. Remote Sens. Environ. **134**, 100–110 (2013)
28. Chen, L.: A study of applying genetic programming to reservoir trophic state evaluation using remote sensor data. Int. J. Remote Sens. **24**(11), 2265–2275 (2003)
29. Chion, C., Landry, J.-A., Da Costa, L.: A genetic-programming-based method for hyperspectral data information extraction: agricultural applications. IEEE Trans. Geosci. Remote Sens. **46**(8), 2446–2457 (2008)
30. Chuvieco, E.: Fundamentals of Satellite Remote Sensing. CRC Press, Jan. 2020
31. Costa, L., Nunes, L., Ampatzidis, Y.: A new visible band index (vNDVI) for estimating NDVI values on RGB images utilizing genetic algorithms. Comput. Electron. Agric. **172**, 105334 (2020)
32. da Silva, J.M., Figueiredo, A., Cunha, J., Eiras-Dias, J.E., Silva, S., Vanneschi, L., Mariano, P.: Using rapid chlorophyll fluorescence transients to classify vitis genotypes. Plants **9**(2), 174, (2020)
33. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans. Evol. Comput. **6**(2), 182–197 (2002)
34. Deb, K.: Multi-objective optimisation using evolutionary algorithms: an introduction. In: Multi-objective Evolutionary Optimisation for Product Design and Manufacturing, pp. 3–34. Springer, London (2011)
35. Domingos, P.: A few useful things to know about machine learning. Commun. ACM **55**(10), 78–87 (2012)
36. Dominic, P., Leahy, D., Willis, M.: GPTIPS:an open source genetic programming toolbox for multigene symbolic regression. Lect. Notes Eng. Comput. Sci. **2180**, 12 (2010)
37. dos Santos, J.A., Ferreira, C.D., da Torres, R.S., Gonçalves, M.A., Lamparelli, R.A.C.: A relevance feedback method based on genetic programming for classification of remote sensing images. Inf. Sci. **181**(13), 2671–2684 (2011)
38. Dubey, A., Inoue, A.H., Birmann, P.T.F., da Silva, S.R.: Evolutionary feature selection. In: Proceedings of the Genetic and Evolutionary Computation Conference. ACM, July 2022
39. Eberhart, R., Kennedy, J.: A new optimizer using particle swarm theory. In: MHS'S95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science. IEEE (1995)
40. Eklund, N.H.W.: Using genetic algorithms to estimate confidence intervals for missing spatial data. IEEE Trans. Syst. Man Cybern. Part C (Applications and Reviews) **36**(4), 519–523 (2006)
41. An experimental study: Enríquez-Zárate, J., Trujillo, L., de Lara, S., Castelli, M., Z-Flores, E., Muñoz, L., Popovič, A.: Automatic modeling of a gas turbine using genetic programming. Appl. Soft Comput. **50**, 212–222 (2017)
42. ESA. Sentinel 2 User Guide, Accessed 31 Aug. 2022. https://sentinels.copernicus.eu/web/sentinel/user-guides/sentinel-2-msi/resolutions/spatial
43. ESA. Sentinel 1 User Guide, Accessed 22 May 2023. https://sentinels.copernicus.eu/web/sentinel/user-guides/sentinel-1-sar
44. García, J.C.F., Kalenatic, D., Bello, C.A.L.: Missing data imputation in multivariate data by evolutionary algorithms. Comput. Hum. Behav. **27**(5), 1468–1474 (2011). 2009 Fifth International Conference on Intelligent Computing
45. Ghimire, S., Deo, R.C., Downs, N.J., Raj, N.: Self-adaptive differential evolutionary extreme learning machines for long-term solar radiation prediction with remotely-sensed MODIS satellite and reanalysis atmospheric products in solar-rich cities. Remote Sens. Environ. **212**, 176–198 (2018)

46. Giustolisi, O., Savic, D.A.: A symbolic data-driven technique based on evolutionary polynomial regression. J. Hydroinf. **8**(3), 207–222 (2006)
47. Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press (2016). http://www.deeplearningbook.org
48. Hart, E., Sim, K., Gardiner, B., Kamimura, K.: A hybrid method for feature construction and selection to improve wind-damage prediction in the forestry sector. In: Proceedings of the Genetic and Evolutionary Computation Conference, New York, NY, USA, July 2017. ACM (2017)
49. Hasan, M.K., Alam, M.A., Roy, S., Dutta, A., Jawad, M.T., Das, S.: Missing value imputation affects the performance of machine learning: a review and analysis of the literature (2010–2021). Inf. Med. Unlock. **27**, 100799 (2021)
50. Hashim, H., Latif, Z.A., Adnan, N.A.: Urban vegetation classification with NDVI threshold value method with very high resolution (VHR) pleiades imagery. Int. Archiv. Photogramm. Remote Sens. Spat. Inf. Sci. XLII-4/W16, 237–240, Oct. 2019
51. He, C., Zhang, Y., Gong, D.: A pseudo-label guided artificial bee colony algorithm for hyperspectral band selection. Remote Sens. **12**(20), 3456 (2020)
52. Holland, J.H.: Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology Control and Artificial Intelligence. MIT Press, Cambridge, MA, USA (1992)
53. Humies. Annual "Humies" Awards For Human-Competitive Results. Accessed 22 May 2023. https://www.human-competitive.org/
54. Ingalalli, V., Silva, S., Castelli, M., Vanneschi, L.: A multi-dimensional genetic programming approach for multi-class classification problems. In: Lecture Notes in Computer Science, pp. 48–60. Springer, Berlin Heidelberg (2014)
55. Johari, A., Habibagahi, G., Ghahramani, A.: Prediction of soil–water characteristic curve using genetic programming. J. Geotech. Geoenviron. Eng. **132**(5), 661–665 (2006)
56. Kabiri, P., Pandi, M.H., Nejat, S.K., Ghaderi, H.: NDVI optimization using genetic algorithm. In: 2011 7th Iranian Conference on Machine Vision and Image Processing. IEEE, Nov. 2011
57. Kanan, H.R., Faez, K., Taheri, S.M.: Feature selection using ant colony optimization (ACO): a new method and comparative study in the application of face recognition system. In: Perner, P. (ed.) Advances in Data Mining. Theoretical Aspects and Applications, pp. 63–76. Springer, Berlin, Heidelberg (2007)
58. Kautsky, H., Hirsch, A.: Neue versuche zur kohlensäureassimilation. Die Naturwissenschaften **19**(48), 964–964 (1931)
59. Keijzer, M., Baptist, M., Babovic, V., Uthurburu, J.R.: Determining equations for vegetation induced resistance using genetic programming. In: Proceedings of the 7th Annual Conference on Genetic and Evolutionary Computation, GECCO '05, pp. 1999–2006, New York, NY, USA (2005). Association for Computing Machinery
60. Key, C., Benson, N.: Landscape Assessment: Ground measure of severity, the Composite Burn Index; and Remote sensing of severity, the Normalized Burn Ratio, pp. LA 1–51. USDA Forest Service, Rocky Mountain Research Station, Jan. 2006
61. Kim, D.-K., Cao, H., Jeong, K.-S., Recknagel, F., Joo, G.-J.: Predictive function and rules for population dynamics of microcystis aeruginosa in the regulated nakdong river (South Korea), discovered by evolutionary algorithms. Ecol. Model. **203**(1–2), 147–156 (2007)
62. Koza, J.R.: Human-competitive results produced by genetic programming. Gen. Programm. Evolvable Mach. **11**(3–4), 251–284 (2010)
63. La Cava, W., Helmuth, T., Spector, L., Moore, J.H.: A probabilistic and multi-objective analysis of lexicase selection and ε-lexicase selection. Evol. Comput. **27**(3), 377–402 (2019)
64. La Cava, W., Moore, J.H.: Semantic variation operators for multidimensional genetic programming. In: Proceedings of the Genetic and Evolutionary Computation Conference. ACM, July 2019
65. La Cava, W., Moore, J.H.: Learning feature spaces for regression with genetic programming. Genet. Program. Evolvable Mach. **21**(3), 433–467 (2020)

66. La Cava, W., Silva, S., Danai, K., Spector, L., Vanneschi, L., Moore, J.H.: Multidimensional genetic programming for multiclass classification. Swarm Evol. Comput. **44**, 260–272 (2019)
67. Lary, D.J., Alavi, A.H., Gandomi, A.H., Walker, A.L.: Machine learning in geosciences and remote sensing. Geosci. Front. **7**(1), 3–10 (2016)
68. Leardi, R., Boggia, R., Terrile, M.: Genetic algorithms as a strategy for feature selection. J. Chemom. **6**(5), 267–281 (1992)
69. Lensen, A., Al-Sahaf, H., Zhang, M., Verma, B.: Genetic programming for algae detection in river images. In: 2015 IEEE Congress on Evolutionary Computation (CEC). IEEE, May 2015
70. Li, X., Mabu, S., Zhou, H., Shimada, K., Hirasawa, K.: Genetic network programming with estimation of distribution algorithms for class association rule mining in traffic prediction. In: IEEE Congress on Evolutionary Computation. IEEE, July 2010
71. Liu, G., Li, Y., Jiao, L., Chen, Y., Shang, R.: Multiobjective evolutionary algorithm assisted stacked autoencoder for PolSAR image classification. Swarm Evol. Comput. **60**, 100794 (2021)
72. Liu, H., Motoda, H.: Feature Extraction, Construction and Selection: A Data Mining Perspective. Kluwer Academic Publishers, USA (1998)
73. Lobato, F., Sales, C., Araujo, I., Tadaiesky, V., Dias, L., Ramos, L., Santana, A.: Multi-objective genetic algorithm for missing data imputation. Pattern Recogn. Lett. **68**, 126–131 (2015)
74. Lotz, M., Silva, S.: Application of genetic programming classification in an industrial process resulting in greenhouse gas emission reductions. In: Applications of Evolutionary Computation, pp. 131–140. Springer, Berlin, Heidelberg (2010)
75. Miao, L., Bi, Y., Xue, B., Qiong, H., Zhang, M., Wei, Y., Yang, P., Wenbin, W.: Genetic programming for high-level feature learning in crop classification. Remote Sens. **14**(16), 3982 (2022)
76. Makkeasorn, A., Chang, N.-B., Li, J.: Seasonal change detection of riparian zones with remote sensing images and genetic programming in a semi-arid watershed. J. Environ. Manag. **90**(2), 1069–1080 (2009)
77. Martínez-Ballesteros, M., Martínez-Álvarez, F., Troncoso, A., Riquelme, J.C.: An evolutionary algorithm to discover quantitative association rules in multidimensional time series. Soft Comput. **15**(10), 2065–2084 (2011)
78. Martínez-Ballesteros, M., Troncoso, A., Martínez-Álvarez, F., Riquelme, J.C.: Mining quantitative association rules based on evolutionary computation and its application to atmospheric pollution. Integr. Comput.-Aided Eng. **17**(3), 227–242 (2010)
79. Mateus, P., Fernandes, P.M.: Forest fires in Portugal: Dynamics, causes and policies. In: Forest Context and Policies in Portugal, pp. 97–115. Springer International Publishing (2014)
80. McConaghy, T.: FFX: Fast, Scalable, Deterministic Symbolic Regression Technology, pp. 235–260. Springer, New York, NY (2011)
81. Mei, Y., Chen, Q., Lensen, A., Xue, B., Zhang, M.: Explainable artificial intelligence by genetic programming: a survey. IEEE Trans. Evol. Comput. 1–1 (2022)
82. Mejia-Zuluaga, P.A., Dozal, L., Valdiviezo-N, J.C.: Genetic programming approach for the detection of mistletoe based on UAV multispectral imagery in the conservation area of mexico city. Remote Sens. **14**(3), 801 (2022)
83. Miller, G.T., Spoolman, S.: Environmental Science, 13th ed. Wadsworth Publishing, Belmont, CA, Jan. 2010
84. Moraglio, A., Krawiec, K., Johnson, C.G.: Geometric semantic genetic programming. In: Lecture Notes in Computer Science, pp. 21–31. Springer, Berlin, Heidelberg (2012)
85. Muñoz, L., Silva, S., Trujillo, L.: M3GP–multiclass classification with GP. In: Lecture Notes in Computer Science, pp. 78–91. Springer International Publishing (2015)
86. Muñoz, L., Trujillo, L., Silva, S.: Transfer learning in constructive induction with genetic programming. Genetic Programm. Evol. Mach. **21**(4), 529–569 (2019)
87. Muñoz, L., Trujillo, L., Silva, S., Castelli, M., Vanneschi, L.: Evolving multidimensional transformations for symbolic regression with M3GP. Memetic Comput. **11**(2), 111–126 (2019)

88. Mustafa, M.T., Hassoon, K.I., Hussain, H.M., Abd, M.H.: Using water indices (NDWI, MNDWI, NDMI, WRI and AWEI) to detect physical and chemical parameters by apply remote sensing and GIS techniques. Int. J. Res.—Granthaalayah **5**(10), 117–128 (2017)
89. Nagkoulis, N., Loukogeorgaki, E., Ghislanzoni, M.: Genetic algorithms-based optimum PV site selection minimizing visual disturbance. Sustainability **14**(19) (2022)
90. NASA. Landsat 8 Bands. Accessed 25 Aug. 2022. https://landsat.gsfc.nasa.gov/landsat-8/landsat-8-bands
91. Olson, R.S., Bartley, N., Urbanowicz, R.J., Moore, J.H.: Evaluation of a tree-based pipeline optimization tool for automating data science. In: Proceedings of the Genetic and Evolutionary Computation Conference 2016, GECCO '16, pp. 485–492, New York, NY, USA. ACM (2016)
92. Omran, M.G.H., Salman, A., Engelbrecht, A.P.: Self-adaptive differential evolution. In: Computational Intelligence and Security, pp. 192–199. Springer, Berlin, Heidelberg (2005)
93. Parasuraman, K., Elshorbagy, A., Si, B.C.: Estimating saturated hydraulic conductivity using genetic programming. Soil Sci. Soc. Am. J. **71**(6), 1676–1684 (2007)
94. Pei, W., Xue, B., Shang, L., Zhang, M.: New fitness functions in genetic programming for classification with high-dimensional unbalanced data. In: 2019 IEEE Congress on Evolutionary Computation (CEC), pp. 2779–2786. IEEE (2019)
95. Pereira, S.C., Lopes, C., Pedroso, J.P.: Mapping cashew orchards in cantanhez national park (Guinea-Bissau). Remote Sens. Appl.: Soc. Environ. **26**, 100746 (2022)
96. Poli, R., Langdon, W.B., Mcphee, N.: A Field Guide to Genetic Programming. Lulu Enterprises ltd., UK, Jan. 2008
97. Puente, C., Olague, G., Smith, S.V., Bullock, S.H., Hinojosa-Corona, A., González-Botello, M.A.: A genetic programming approach to estimate vegetation cover in the context of soil erosion assessment. Photogramm. Eng. Remote Sens. **77**(4), 363–376 (2011)
98. Quade, M., Abel, M., Shafi, K., Niven, R.K., Noack, B.R.: Prediction of dynamical systems by symbolic regression. Phys. Rev. E 94(1), (2016)
99. Rodrigues, N.M., Batista, J.E., La Cava, W., Vanneschi, L., Silva, S.: Slug: feature selection using genetic algorithms and genetic programming. In: Medvet, E., Pappa, G., Xue, B. (eds.) Genetic Programming, Cham, pp. 68–84. Springer International Publishing (2022)
100. Rodrigues, N.M., Batista, J.E., Silva, S.: Ensemble genetic programming. In: Hu, T., Lourenço, N., Medvet, E., Divina, F. (eds.), Genetic Programming, Cham, pp. 151–166. Springer International Publishing (2020)
101. Rolnick, D., Donti, P.L., Kaack, L.H., Kochanski, K., Lacoste, A., Sankaran, K., Ross, A.S., Milojevic-Dupont, N., Jaques, N., Waldman-Brown, A., Luccioni, A.S., Maharaj, T., Sherwin, E.D., Mukkavilli, K., Kording, K.P., Gomes, C.P., Ng, A.Y., Hassabis, D., Platt, J.C., Creutzig, F., Chayes, J., Bengio, Y.: Tackling climate change with machine learning. ACM Comput. Surv., **55**(2) (2022)
102. Rouse, J.W., Haas, R.H., Schell, J.A., Deering, D.W.: Monitoring vegetation systems in the great plains with ERTS. In: NASA Special Publication (1973)
103. Rudin, C.: Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. Nat. Mach. Intell. **1**(5), 206–215 (2019)
104. Saxena, N.K., Kumar, A.: Reactive power control in decentralized hybrid power system with STATCOM using GA, ANN and ANFIS methods. Int. J. Electr. Power Energy Syst. **83**, 175–187 (2016)
105. Semet, Y., Berthelot, B., Glais, T., Isbérie, C., Varest., A.: Expert competitive traffic light optimization with evolutionary algorithms. In: Proceedings of the 5th International Conference on Vehicle Technology and Intelligent Transport Systems. SCITEPRESS—Science and Technology Publications (2019)
106. Shang, Y., Zheng, X., Li, J., Liu, D., Wang, P.: A comparative analysis of swarm intelligence and evolutionary algorithms for feature selection in SVM-based hyperspectral image classification. Remote Sens. **14**(13), 3019 (2022)
107. Silva, S., Dignum, S., Vanneschi, L.: Operator equalisation for bloat free genetic programming and a survey of bloat control methods. Gen. Programm. Evol. Mach. **13**(2), 197–238 (2011)

108. Silva, S., Ingalalli, V., Vinga, S., Carreiras, J.M.B., Melo, J.B., Castelli, M., Vanneschi, L., Gonçalves, I., Caldas, J.: Prediction of forest aboveground biomass: an exercise on avoiding overfitting. In: Applications of Evolutionary Computation, pp. 407–417. Springer Berlin Heidelberg (2013)
109. Silva, S., Muñoz, L., Trujillo, L., Ingalalli, V., Castelli, M., Vanneschi, L.: Multiclass classification through multidimensional clustering. In: Genetic Programming Theory and Practice XIII, pp. 219–239. Springer International Publishing (2016)
110. Silva, S., Tseng, Y.-T.: Classification of seafloor habitats using genetic programming. In: Lecture Notes in Computer Science, pp. 315–324. Springer, Berlin, Heidelberg (2008)
111. Silva, S., Vanneschi, L., Cabral, A.I.R., Vasconcelos, M.J.: A semi-supervised genetic programming method for dealing with noisy labels and hidden overfitting. Swarm Evol. Comput. **39**, 323–338 (2018)
112. Silva, S., Vasconcelos, M.J., Melo, J.B.: Bloat free genetic programming versus classification trees for identification of burned areas in satellite imagery. In: Applications of Evolutionary Computation, pp. 272–281. Springer, Berlin, Heidelberg (2010)
113. Stanislawska, K., Krawiec, K., Kundzewicz, Z.W.: Modeling global temperature changes with genetic programming. Comput. Math. Appl. **64**(12), 3717–3728 (2012)
114. Stolfi, D.H., Alba, E.: Greener routes with bio-inspired techniques. Green swarm. Appl. Soft Comput. **71**, 952–963 (2018)
115. Taghizadeh-Mehrjardi, R., Ayoubi, S., Namazi, Z., Malone, B.P., Zolfaghari, A.A., Sadrabadi, F.R.: Prediction of soil surface salinity in arid region of central Iran using auxiliary variables and genetic programming. Arid Land Res. Manag. **30**(1), 49–64 (2016)
116. Back, T.: Evolutionary Algorithms in Theory and Practice. Oxford University Press, New York, NY (1996)
117. Trabucchi, M., Puente, C., Comin, F.A., Olague, G., Smith, S.V.: Mapping erosion risk at the basin scale in a mediterranean environment with opencast coal mines to target restoration actions. Reg. Environ. Change **12**(4), 675–687 (2012)
118. Tran, B., Xue, B., Zhang, M.: Genetic programming for feature construction and selection in classification on high-dimensional data. Memet. Comput. **8**(1), 3–15 (2015)
119. Tsanas, A., Xifara, A.: Accurate quantitative estimation of energy performance of residential buildings using statistical machine learning tools. Energy Build. **49**, 560–567 (2012)
120. Vadillo, J., Santana, R., Lozano, J.A.: When and how to fool explainable models (and humans) with adversarial examples (2021)
121. Vasconcelos, M., Silva, S., Tomé, M., Alvim, M., Pereira, J.: Spatial prediction of fire ignition probabilities: comparing logistic regression and neural networks. Photogram. Eng. Remote Sens. **67**, 73–81, 01 (2001)
122. Viegas, F., Rocha, L., Gonçalves, M., Mourão, F., Sá, G., Salles, T., Andrade, G., Sandin, I.: A genetic programming approach for feature selection in highly dimensional skewed data. Neurocomputing **273**, 554–569 (2018)
123. Wen, C., Miao, L., Bi, Y., Zhang, S., Xue, B., Zhang, M., Zhou, Q., Wenbin, W.: An object-based genetic programming approach for cropland field extraction. Remote Sens. **14**(5), 1275 (2022)
124. Wen, F., Zhang, G., Sun, L., Wang, X., Xiaowei, X.: A hybrid temporal association rules mining method for traffic congestion prediction. Comput. Ind. Eng. **130**, 779–787 (2019)
125. Yue, W., Ma, W., Miao, Q., Wang, S.: Multimodal continuous ant colony optimization for multisensor remote sensing image registration with local search. Swarm Evol. Comput. **47**, 89–95 (2019)
126. Yang, H., Du, Q.: Particle swarm optimization-based dimensionality reduction for hyperspectral image classification. In: 2011 IEEE International Geoscience and Remote Sensing Symposium. IEEE (2011)
127. Yang, Y., Xue, B., Jesson, L., Zhang, M.: Genetic programming for symbolic regression: a study on fish weight prediction. In: 2021 IEEE Congress on Evolutionary Computation (CEC). IEEE, June 2021

128. Zhang, Q.-S., Zhu, S.-C.: Visual interpretability for deep learning: a survey. Front. Inf. Technol. Electron. Eng. **19**(1), 27–39 (2018)
129. Zhang, Yu., Tiňo, P., Leonardis, A., Tang, K.: A survey on neural network interpretability. IEEE Trans. Emerging Top. Comput. Intell. **5**(5), 726–742 (2021)
130. Zhang, Y., Zhou, Y., Jiang, S., Ning, S., Jin, J., Cui, Y., Wu, Z., Feng, H.: A simulation study using machine learning and formula methods to assess the soybean groundwater contribution in a drought-prone region. Water **14**(19) (2022)