



Article

Application of Supervised Machine Learning Technique on LiDAR Data for Monitoring Coastal Land Evolution

Maurizio Barbarella ¹, Alessandro Di Benedetto ^{2,*} and Margherita Fiani ²¹ DICAM-ARCES, University of Bologna, 40136 Bologna, Italy; maurizio.barbarella@unibo.it² Department of Civil Engineering, University of Salerno, 84084 Fisciano, Italy; m.fiani@unisa.it

* Correspondence: adibenedetto@unisa.it

Abstract: Machine Learning (ML) techniques are now being used very successfully in predicting and supporting decisions in multiple areas such as environmental issues and land management. These techniques have also provided promising results in the field of natural hazard assessment and risk mapping. The aim of this work is to apply the Supervised ML technique to train a model able to classify a particular gravity-driven coastal hillslope geomorphic model (slope-over-wall) involving most of the soft rocks of Cilento (southern Italy). To train the model, only geometric data have been used, namely morphometric feature maps computed on a Digital Terrain Model (DTM) derived from Light Detection and Ranging (LiDAR) data. Morphometric maps were computed using third-order polynomials, so as to obtain products that best describe landforms. Not all morphometric parameters from literature were used to train the model, the most significant ones were chosen by applying the Neighborhood Component Analysis (NCA) method. Different models were trained and the main indicators derived from the confusion matrices were compared. The best results were obtained using the Weighted k-NN model (accuracy score = 75%). Analysis of the Receiver Operating Characteristic (ROC) curves also shows that the discriminating capacity of the test reached percentages higher than 95%. The model, resulting more accurate in the training area, will be extended to similar areas along the Tyrrhenian coastal land.

Keywords: landslides; morphometric analysis; computer vision; supervised learning; Neighborhood Component Analysis; Receiver Operating Characteristic; Digital Terrain Model; LiDAR data



Citation: Barbarella, M.; Di Benedetto, A.; Fiani, M. Application of Supervised Machine Learning Technique on LiDAR Data for Monitoring Coastal Land Evolution. *Remote Sens.* **2021**, *13*, 4782. <https://doi.org/10.3390/rs13234782>

Academic Editors: Mercedes E. Paoletti and Juan M. Haut

Received: 20 October 2021
Accepted: 22 November 2021
Published: 25 November 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Landslide-prone areas' identification and classification play an important role in land assessment, planning and management. They are usually performed using a supervised approach, either through direct geomorphological analysis [1] or through visual interpretation of optical images, e.g., satellite (panchromatic, multispectral) or from Unmanned Aerial Vehicles (UAV) [2,3], or even from Digital Terrain Models (DTM) derived from Light Detection and Ranging (LiDAR) survey [4–6]. Radar or Synthetic Aperture Radar (SAR) imagery is also used to detect acting deformation; interferometric techniques allow the derivation of multi-temporal surface deformation maps with high accuracy and spatial resolution [7].

Although visual interpretation is highly reliable [5,6], the process, besides being subjective, is both time-consuming and labor-intensive. Therefore, automated or semi-automated methods for landslide identification based on remote sensing techniques have been studied greatly in recent years; several automated techniques for landslide susceptibility mapping have been proposed in the past two decades [8,9]. The exponential increase in the number of suggested methods is related to continued advances in computer technology, including the development of increasingly high-performance algorithms and computing processors and increased storage capacity.

A number of models used to study landslide susceptibility exist; they can be roughly divided into two main categories, physical-based models and empirical, data-driven

models [10]. Physical-based models analyze susceptibility using soil and rock mechanics; the lack or dearth of geotechnical data is still a major liability in physical models. Empirical models, instead, mainly use remote sensing-based information, such as satellite imagery and high-resolution DTM, land cover and precipitation products; in this way, large-scale predictions can be conducted and also where not enough geotechnical information is available to apply physical models [11]. All these models are “static” and time-dependent, as they do not consider the progressive evolution from landslide-prone area to landslide-affected area. Empirical models, which are data-driven, mainly include three types of models: heuristic [12], general statistical [13] and Machine Learning (ML) models [9].

ML is seen as a part of Artificial Intelligence (AI) and is based on algorithms that can “learn” from experience, through automated learning (Supervised or Unsupervised). The ML aims to learn to automatically recognize complex patterns and be able to take “thinking” decisions based on the acquired data. Models, “trained” to recognize a given situation on areas with known characteristics, can be used to make predictions on different areas of unknown nature. Many kinds of ML algorithms have been developed and applied to produce landslide susceptibility maps in different regions of the world, the most used are: binary logical regression (BLR) and artificial neural networks (ANNs) [14], fuzzy logic [15], decision tree (DT) [16], random forest (RF) [17], support vector machine (SVM) [18], Bayesian network [19], neuro-fuzzy algorithm [20] and Naïve Bayes algorithm [21]. As highlighted, a variety of different ML models have been used, although a most accurate model has not yet been defined [22].

In recent years, very good results have been obtained using methods based on object-oriented analysis, ANNs and SVMs [9]. Most studies show that in the context of landslide assessment, the solution tends to be nonlinear, due to the complexity of the geological environment as well as factors related to the triggering itself. The unavoidable presence of spatial autocorrelation among the input data, if not taken into account, could lead to erroneous results [23].

Usually, AI-based techniques have proven to be a powerful and promising tool in many engineering applications related to landslide identification [8]. The latest AI algorithms are capable of handling large and complex datasets with the ability to obtain predictions with high success rates, especially when based on methods that avoid overfitting such as k-fold cross-validation [24].

Given the sudden development of remote sensing sensors and the chance to get high-resolution data, even open source, in record time, several research papers have been written on ML for studying landslide susceptibility mapping (LSM) and automatic landslide mapping (ALM), which are mainly based on the use of variables derived from remote sensed data [25,26].

X. Chen and W. Chen [27] evaluated the spatial prediction of landslides using bivariate statistical-based kernel logistic regression machine learning classifiers, starting from fourteen landslide conditioning factors (LCF) analyzed via multicollinearity analysis; morphometric parameters of geometric nature (derived from DTM), geological (e.g., lithology, soil, land use) and indices derived from multispectral analysis, such as the NDVI vegetation index, have been used.

The trained models result accurate, and the landslide susceptibility maps could be successfully used by government agencies for prevention and mitigation of hydrogeological risk. Youssef and Pourghasemi [28] performed a study to compare the results of seven ML algorithms (SVM, RF, ANN, Quadratic Discriminant Analysis, Linear Discriminant Analysis, Naive Bayes and Multivariate Adaptive Regression Spline); for training, the authors use remotely sensed data, geological data, and other conditioning factors such as vegetation and land-use indices. The accuracy achieved confirms that the largest contribution comes from geometric parameters derived from remotely sensed data.

Novellino et al. [29] used the Generalized Boosting Model (ANN) technique to derive the Landslide Risk (LR) of a coastal area in Southern Italy by using variables derived from remotely sensed data (radar, LiDAR and satellite images), as well as from geological

and geomorphological data. The authors also introduced an innovative method based on the combination of data coming from the InSAR technique and Ensemble Modeling (EM); again, the results show that the variables that most influence the process for the determination of landslide risk are the geometric ones derived from remotely sensed data.

Nsengiyumva and Valentino [30] used three distinct machine learning methods in a GIS environment to map landslide susceptibility; RF, Naïve Bayes Tree (NBT) and Logistic Model Tree models were trained and compared. The predictors used are based on features derived from processing of remotely sensed data (DTM, NDVI) as well as geologic land use and hydrologic data. The NBT model produced more accurate results in terms of discrimination ability; overall, the three methods result, for the analyzed area, all accurate and promising.

Merghadi et al. [25] presented a detailed overview of the ML methods developed in the last two decades about the analysis of landslide susceptibility, comparing the main ML algorithms. The authors highlighted that the relevant parameters to be considered in training are mainly function of the scale of the analysis and the case study; they should also be chosen according to their influence on the triggering mechanisms [31]. In this study, too, the hydrological and geomorphometric parameters are the ones that provide the greatest contribution to the result, an aspect in line with the recent works cited and the most relevant studies referred to in the literature.

Choosing the proper spatial resolution of the DTM is a key issue for an effective landslide analysis based on data acquired with remote sensing techniques. Several studies have shown that the highest resolution is not always the best solution [32,33].

Pawluszek et al. [34] presented a Pixel-Based study on the sensitivity analysis of ALM; the authors assessed the influence on the classification results of the resolution of the LiDAR-based DTM and analyzed different morphological indicators (based on Aspect, Topographic Position Index, Slope, Curvature and Roughness), computed with different kernel sizes, to evaluate their impact on the classification algorithms used: Maximum Likelihood, Feed-Forward Neural Networking (as in ANNs) and SVM. The morphological indicators influence differently the classification algorithms depending on the resolution of the DTM from which they derive. Feature sensitivity, for selected kernel sizes, increases with coarser DTM resolution. The authors also suggested future research to study models able to classify landslides according to type and size.

In a few studies regarding ALM, only DTM-based features have been used [35,36]; in others, they have been integrated with other remotely sensed data [37,38]; here also, the key aspect is the choice of the parameters (predictors) used for classification [37,39,40].

The literature review also showed that there is no ML model that can be defined as the most suitable for a specific problem, therefore, selecting the most suitable method for landslide spatial prediction does not depend solely on the underlined scientific goal for the case study [41,42].

Research papers on the use of ML for the study of models that can classify landslide-prone or susceptible areas are many, but all authors highlighted the need for further research to develop methodologies that minimize subjectivity in the selection of input data for training, since there are no universal rules for selecting conditioning factors for classification or for creating landslide susceptibility maps [43,44].

The aim of our work is to use the Supervised ML (supervised learning) technique to identify and classify a specific coastal land evolution model (slope-over-wall) that characterizes most of the Cilento soft rock coasts [45]. This geomorphological phenomenon is characterized by the presence of three contiguous areas having recognizable and previously studied geomorphometric features [6].

DTM can be effective in representing specific landforms accurately; however, the potential of landslide identification using machine learning and deep learning from DTM and its derivatives has not yet been widely exploited [46].

Most of the studies using DTM in classification algorithms also use non-geometric data as input, e.g., data of hygrometric nature, lithology, indices, etc. Many research papers

show that the morphometric parameters extracted from DTM are those that provide a greater contribution to achieve a good result. The goal of our work is to use, in training classification models, only predictors of geometric nature, i.e., morphometric parameters derived from high-resolution DTM. Due to the wide range of potential applications, high-precision DTMs are often made available by national or regional authorities, while remotely sensed data should generally be acquired ad hoc.

We analyzed the features (morphometric parameters) most suitable to characterize the area and the models that allow to take full advantage of their characteristics. The model assessed to be more accurate in prediction will be chosen and applied to test other areas, to evaluate the portability and effectiveness of the model on areas with characteristics similar to that of training.

2. Case Study and Materials

The study area is located along the Cilento coast in the Campania Region (southern Italy) (Figure 1). It covers a coastal area, which is affected by landslide phenomena [5]. The coastal area has been described as a “slope-over-wall” model, composed of a convex, colluvial, debris upper slope laying on remnants of a buried, uplifted marine platform covered by rounded, gravelly marine deposits hanging on the cliffed bedrock toe slope.

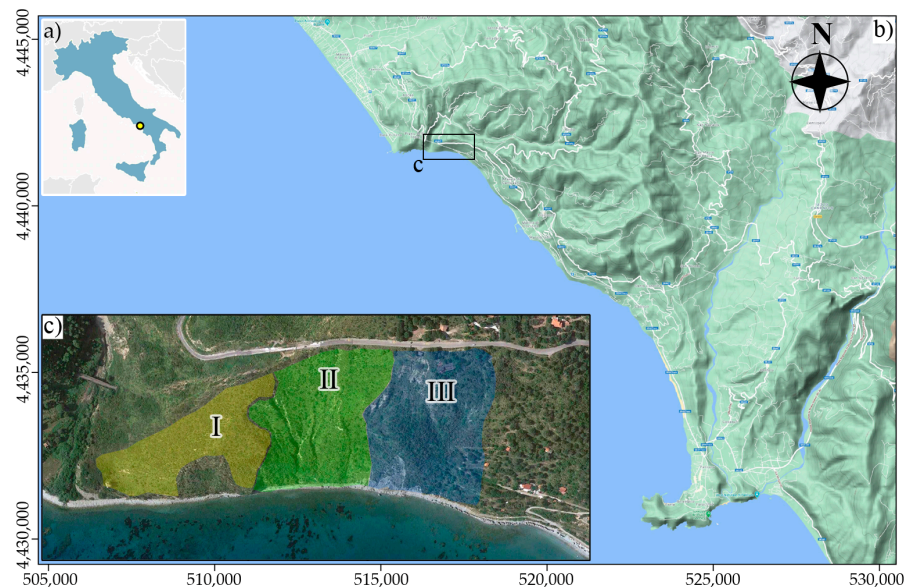


Figure 1. Study area. (a) Map of Italy, the circler identifies the area; (b) A zoomed-in view of the area, the box highlights the training area; (c) The training area, with the three sections highlighted.

The temporal and spatial evolution of this phenomenon can be described by schematizing the geomorphological evolutionary process in three distinct coastal sections, clearly distinguishable [6,45]:

1. Area I, corresponding to the western coastal strip, where the morphology of the original model of “slope-over-wall” is preserved; in terms of morpho-structural evolution, it corresponds to the initial and unperturbed stress stage.
2. Area II, corresponding to the intermediate stage of the evolutionary process; the original cliff is fragmented by gullies and ravines affected by erosive and flow processes triggered by shallow retrogressive landslides.
3. Area III, representing the space-time expression of the definitive gravity-driven evolution of the coastal slope; it corresponds to the area progressively affected by active, reactivated and deep-seated landslides.

Figure 1c shows the three sections described. Each section is characterized by a peculiar geomorphological feature that leads to topographic variations well distinguishable and consequently analyzable through geo-morphometric parameters. Accordingly, some

of these parameters will be derived from the DTM in order to identify a model that best identifies the three coastal sections.

This study area in Figure 1c has been chosen to train the model whereas other coastal areas with similar characteristics have been used for testing.

On the wide area, as well as on the whole Italian territory, LiDAR data from the survey carried out in 2012 on behalf of the Ministry of Environment and Protection of Land and Sea (MATM) are available (<http://www.pcn.minambiente.it/mattm/progetto-pst-dati-lidar/> accessed on 19 October 2021). The data were acquired by an Optech Lidar ALTM System mounted on an aircraft flying at an altitude between 1500 and 1800 m AGL (height above ground level). The direction of flight is parallel to the coastline, the maximum scan angle used is 25° and the scan frequency has been set equal to 100 kHz.

The specifications of the data, as stated by the distributing authority, are as follows: point density greater than 1.5 points per square meter, planimetric accuracy (2σ) of 30 cm, altimetric accuracy (1σ) of 15 cm. The available point clouds are referenced in the current National Reference System (ETRS89/ETRF00).

To test the model, different sites of the Cilento coast were searched to select those possibly characterized by the same “slope-over-wall” phenomenon. This morpho-evolutionary phenomenon was particularly evident in two areas [45]: (i) in the stretch of coastline called “Ripe Rosse”, in northern Cilento and (ii) to the west of the area used as training, in the coastal stretch of the Marina di Ascea reef. The data used for the test belong to the same survey campaign that produced the data used for training. Figure 2 shows the map with the areas used for testing.

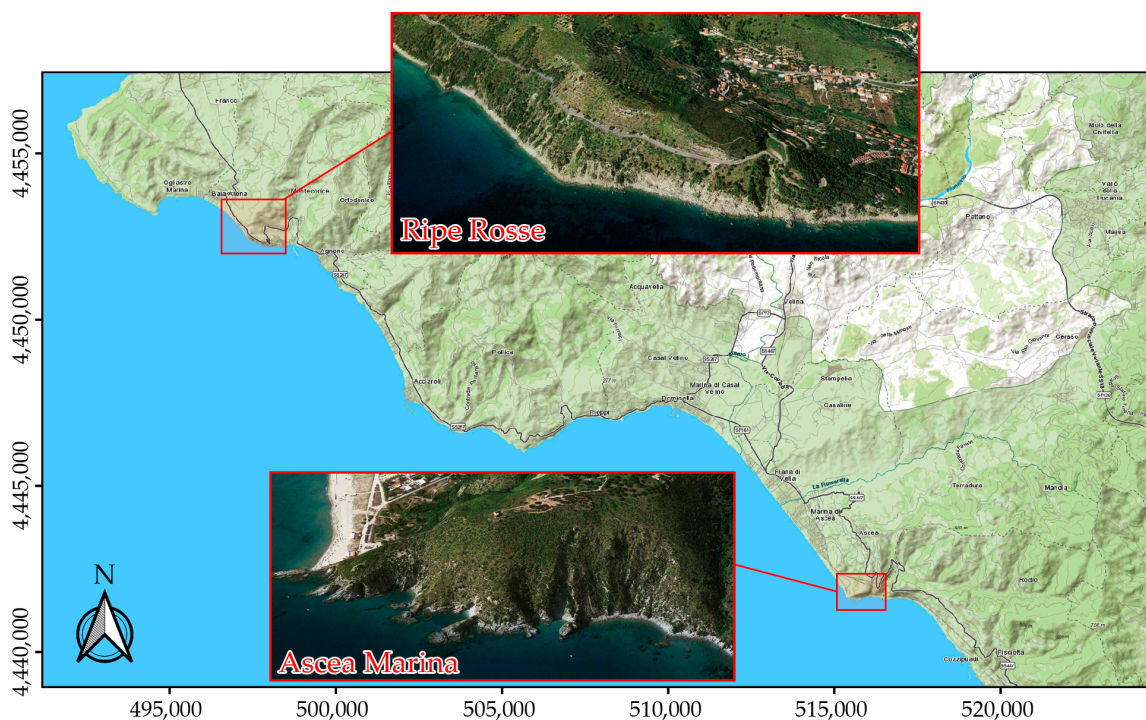


Figure 2. Testing sites (Google Earth maps).

3. Methods

The aim of our work was to study and train a model using the Supervised Machine Learning technique, able to predict the typology of coastal section (I, II, III) using strictly DTM-based geometric morphometric parameters. The LiDAR point cloud was filtered before interpolating the DTM, using the Multiscale Curvature Classification (MCC) filtering algorithm, according to the procedure described in detail in [6].

After having selected the most appropriate supervised model among those evaluated, we checked its classification ability on other data for which the attribution class is not

known a priori. The exploited tool allows to export the trained model, so that it is possible to make predictions using new data. The chosen model is applied to other areas than the one used for training.

The methodology implemented is based on several steps, listed below:

1. Building of a 5 m resolution LiDAR-derived DTM.
2. Computation of morphometric parameters for each individual coastal section.
3. Selection of the morphometric parameters that are deemed significant for our classification problem, using Neighborhood Component Analysis (NCA).
4. Training of a few selected models, their validation and choice of the one providing the best accuracy.
5. Testing of the trained model on two different areas characterized by the same morpho-evolutionary process.

Figure 3 shows the workflow of the methodology used; the whole process is implemented in MATLAB environment.

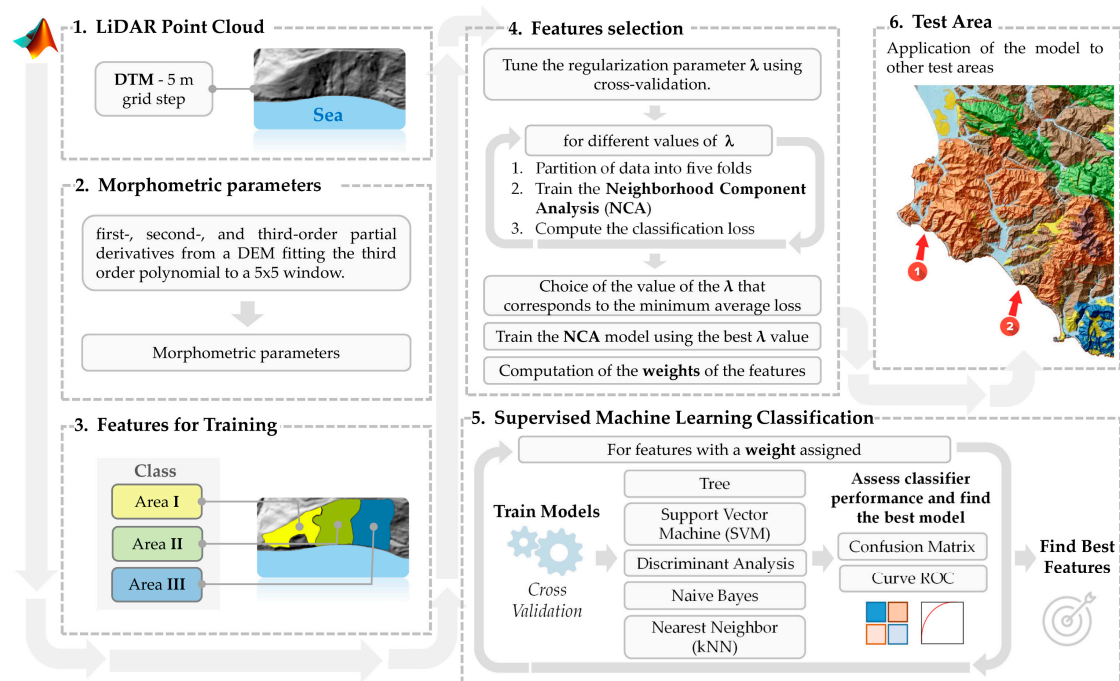


Figure 3. Workflow.

3.1. Maps of Geomorphometric Parameters

The maps of geomorphometric parameters are derived from a DTM built starting from the filtered LiDAR point cloud, interpolated to the nodes of a 5 m grid.

The choice of the optimal pixel size depends on several things, including the complexity of the terrain, the scale of analysis, and the type of landslide to be analyzed [33]. Pawłuszek et al. [32] reported a study (accuracy assessment) on the choice of the optimal pixel size for automatic landslide mapping (ALM), using LiDAR-derived DTM. The authors pointed out that in the case of classification with ML algorithms, there is a close relationship between performance and DTM resolution. When using a resolution of 5 m, an excellent compromise between classification accuracy and processing time is obtained. There is no optimal resolution but rather a range of relevant resolutions [34].

For the spatial interpolation of data and the construction of the DTM, we used the kriging method which, unlike other interpolators, requires the variogram modeling which in turn needs the study of the spatial distribution of the data. Studies on variogram models used for this area can be found in previous papers [5,6].

To choose the morphometric parameters of interest to be derived from the DTM, we referred to the relevant literature on the subject, in particular, we followed the approach proposed by Foroutan et al. [47] who analyzed twenty-two parameters and among them, identified nine as the most significant for the unsupervised classification (self-organizing maps) of landforms, being based on the Optimum Index Factor (OIF). The ones chosen are as follows: 1. difference of curvature (Difc); 2. slope insolation (Slins); 3. rotor (Rot); 4. aspect (Asp); 5. cross-sectional curvature (Cros); 6. total ring curvature (TRc); 7. extreme curvature (Extc); 8. vertical curvature (Verc); 9. unsphericity (Unsph).

From the DTM, nine features have been derived according to the formulas reported in [47]. Curvature, particularly, has important implications for surface processes [48]. It has been shown that the quadratic models [49] can be used to model geomorphometric elements (ridges, slopes, valleys) and basic hillslope units. Higher-order polynomials, which produce non-uniform curvature for the analysis window, can represent special landform features with a more complex structure.

The formulas for computing the first, second, and third-order partial derivatives of the elevation of the DTM nodes are those introduced by Florinsky [50], who used a third-order polynomial instead of a second-order polynomial as proposed by Evans [51].

The method developed by Florinsky has proven to be more accurate in the computation of partial derivatives (in terms of root mean square error), reducing the uncertainty in the computation of the morphometric parameters, thus the derived maps result to be more detailed in the description of the features (shapes) of the ground [50]. The coefficients of the polynomial equation were computed using a 5×5 moving window on the DTM.

3.2. Feature Selection Using Neighborhood Component Analysis

To train the classification model, not all computed features always contribute usefully, so it is advisable to select the truly relevant ones. To select those features that maximize the prediction accuracy of classification, we used the method known as NCA [52], a non-parametric method developed to be used especially in combination with k-nearest neighbor (k-NN) classification models. In details, we used the k-fold cross-validation (k-fold CV) method (with $k = 5$).

The goal of NCA is to maximize the regularized objective function $F(w)$ with respect to the weights [53]:

$$F(w) = \frac{1}{n} \sum_{i=1}^n p_i - \lambda \sum_{r=1}^p w_r^2 \quad (1)$$

where λ is the regularization parameter and p_i is the average Leave-One-Out probability of correct classification of the observation i . There is only one regularization parameter λ for all weights, which can drive some of them to 0.

For selecting features, we used the “fscnca” function in MATLAB, which requires, as input, the matrix containing the predictors, with a number of rows n equal to the number of observations (number of pixels within the coastal sections) and a number of columns p equal to the number of features. In addition, the vector containing the class labels (classes related to the geomorphometric/coastal section, associated with each single observation, row of the matrix containing the predictors or features) is required as input.

As suggested in Ref. [53], $1/n$ could be used as the value of λ but we rather used the one that led to the lowest loss.

The input data, namely predictors and classes, are randomly divided into five sub-sets, of which four are used as training set for the NCA model and the remaining sub-set is used as testing set.

Given a range of values of λ (starting from $\lambda = 1/n$), the k-fold CV was run for any value of λ , so as to compute the associated loss value, with the function “classification loss”, which requires as input the NCA function $F(w)$.

The process of computing the loss value is repeated five times, with the sub-set for testing always different. There will be a total of five loss values, and the one associated with the single input value of λ will be the average of the five. The whole process is run for

all values of λ contained in the range. The optimized regularization parameter λ will be the one with the minimum loss value.

Once the λ value has been estimated, the weights w_i of the features have been also estimated using the technique called Stochastic Gradient Descent (SGD), an effective learning algorithm when the training set is large. Features with a weight below a certain threshold T :

$$T = \tau \cdot \max(w_i) \quad (2)$$

may be rejected, i.e., not used in model training; the tolerance parameter τ suggested in [53] is 0.02.

3.3. Supervised Machine Learning Classification

Supervised Machine Learning is a technique that uses an existing supervised classified dataset as a training dataset to make predictions [54]. The training dataset includes input variables (features) and response variables (in our case, the classes that correspond to coastal sections I, II, III). Machine learning algorithms use computational methods to “learn” information directly from the data without relying on a predetermined equation as a model. The algorithms adaptively improve their performance as the number of available variables for learning increases.

Supervised learning uses classification or regression techniques to develop predictive models, whereas unsupervised learning uses the clustering technique to detect hidden groupings of data. There are multiple supervised machine learning algorithms; complex, highly flexible models usually lead to overfitting of the data, modeling small variations that would be noise. Whereas models with low flexibility generally are easier to interpret but may result in lower accuracy. The model that best represents the input data should be the one that provides a golden mean between accuracy of results and model complexity [55].

To train and select the model that best fits the input data, the MATLAB tool “Classification Learner App” was used, which is a tool that allows to train different models. All the classification models from the relevant literature were trained: Decision Trees (DT), Discriminant Analysis, Support Vector Machines (SVM), k-Nearest Neighbors (k-NN), Naive Bayes, so as to compare them and choose the one that maximizes the overall accuracy.

To assess how well models fit data using cross-validation, we computed the accuracy score, that is, number of correct predictions divided by the total number of input samples, using all observations found in a held-out fold [56].

The model with the highest accuracy score might not be the most ideal model because it might fit better to details of the training sample rather than to its overall trend (overfitting) whereas a model with a slightly lower overall accuracy might be the best classifier for us.

In addition, to analyze the behavior of the model for each class, some main indicators, derived from the confusion matrix, have been computed, namely the True Positive Rate (*TPR*), the False Negative Rate (*FNR*), the Positive Predictive Value (*PPV*) and the False Discovery Rates (*FDR*). In details: the *TPR*, also called “Recall” or “Sensitivity”, is used to measure the percentage of actual positives which are correctly identified; the *FNR* is the remaining percentage of *TPR*; the *PPV*, also called “Precision”, is the proportion of correctly classified observations per predicted class; the *FDR* is the proportion of incorrectly classified observations per predicted class. The formulas used are [57]:

$$\begin{aligned} TPR &= \frac{TP}{TP+FN}; FNR = \frac{FN}{TP+FN} = 1 - TPR; \\ PPV &= \frac{TP}{TP+FP}; FDR = \frac{FP}{FP+TP} = 1 - PPV \\ Accuracy &= \frac{TN+TP}{TN+FP+TP+FN} \end{aligned} \quad (3)$$

where *TP* are the True Positive (samples correctly classified as positive), *TN* are the True Negative (samples correctly classified as negative), *FP* are the False Positive (samples incorrectly classified as positive) and *FN* are the False Negative (samples incorrectly classified as negative).

Often, there is an inverse relationship between Precision (*PPV*) and Sensitivity (*TPR*): when precision increases, model sensitivity worsens and vice versa. For these reasons, it is important to find the golden mean, meaning a balance between the two indicators, so as to obtain a model that best fits the input data.

In addition, for the same purpose of assessing the classification accuracy of each single class, the standard ROC (Receiver Operating Characteristic) method was used [58]. In a ROC curve, the true positive rate (Sensitivity) is plotted as function of the false positive rate (Specificity) for different cut-off points of a parameter.

Each point on the ROC curve represents a sensitivity/specificity pair corresponding to a particular decision threshold. The Area Under the ROC curve (AUC) is a measure of how well a parameter can distinguish between two diagnostic groups (diseased/normal). The ROC curve also allows to find the best cut-off, that is, the test value that maximizes the difference between true positives and false positives.

4. Results

4.1. Morphometric Maps

Morphometric parameters have been computed starting from the DTM derived from LiDAR data filtered for vegetation and artifacts. As for the MCC algorithm used for filtering, the values chosen for the key parameters are $\lambda = 0.55$ and $t = 0.095$. The parameter t used is greater than the default suggested, this is due to the roughness of the terrain of the analyzed area, so using a greater value of t better preserves the natural morphology of the terrain. The scale factor λ is congruent with the density of the point cloud (about 1.5 points/m²).

Figure 4 shows the classified maps of morphometric parameters computed from the DTM at 5 m resolution. For a better visualization, the subdivision of the feature classes was done with the Natural Breaks method.

4.2. Feature Selection

To tune λ , that is, finding the value that will produce the least classification loss for feature selection, and to maximize the prediction accuracy of the NCA classification algorithm using the “fscnca” function, we used five-fold cross-validation.

First, we computed the nine morphometric parameters selected for analysis (Difc, Slins, Rot, Asp, Crosc, TRc, Extc, Verc, Unsph) for each coastal section (I, II, III in Figure 1). These were sorted by columns in a matrix containing a number of rows equal to the number of observations ($n = 4558$) and a number of columns p equal to the number of features used ($p = 9$ features). The column vector containing the class labels (coastal section I, II, III) for each observation also had a number of rows equal to $n = 4558$.

The NCA algorithm was run for twenty values of λ , within the range [0–0.011] in steps of 5.8×10^{-4} . The range of values of λ analyzed was chosen in order to identify the minimum of the function whereas the step chosen is a threshold value, based on tests. The procedure was repeated for the five different folds.

The mean of the five loss values computed for each k-fold is associated with the corresponding λ . The mean values are shown in Figure 5a for each value of λ of the considered range. The smallest loss value, equal to 0.35, is obtained in correspondence of $\lambda = 0.0046$, which is assumed as optimized parameter. In Figure 5b are reported the values of the weights relative to each feature, computed with that value of parameter.

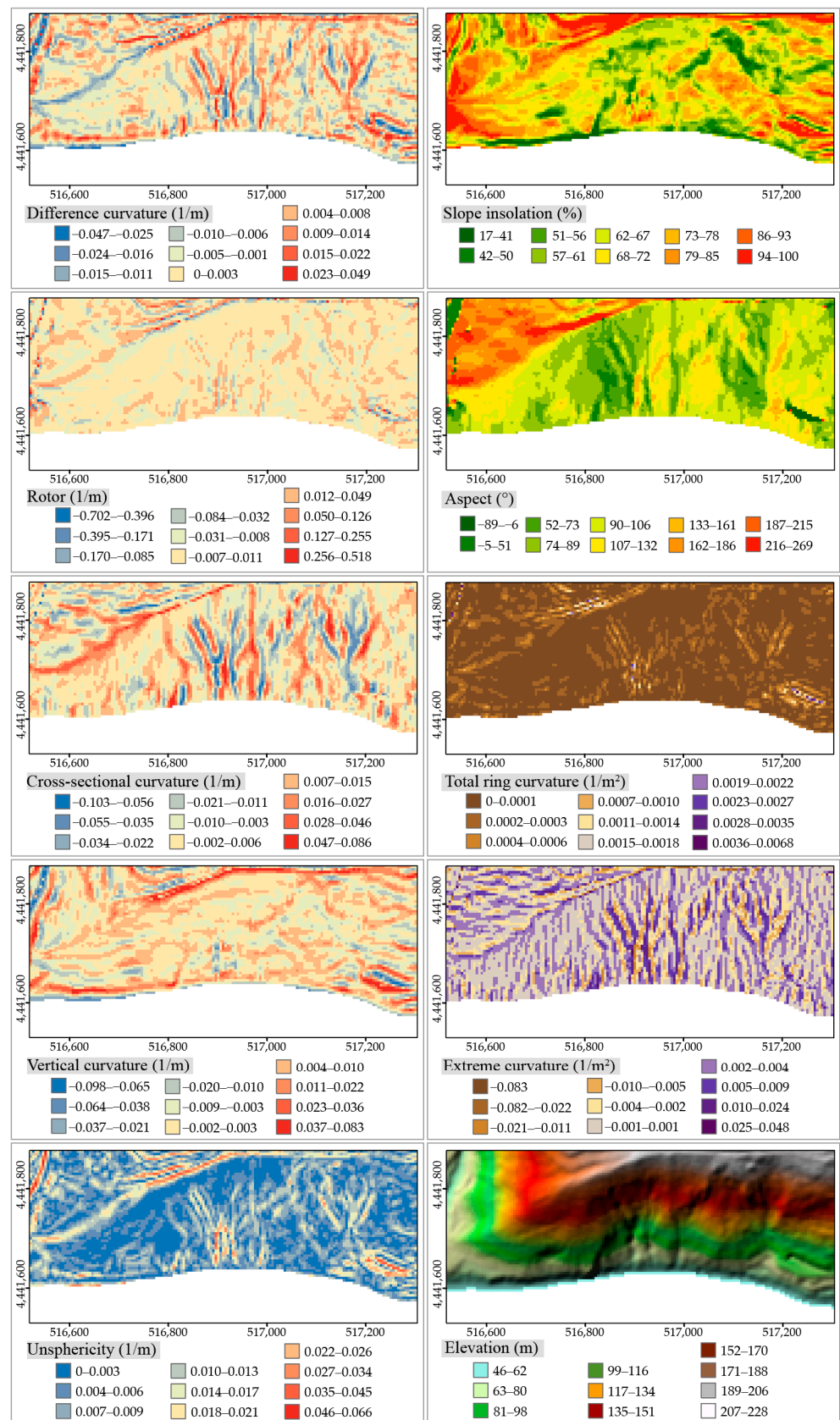


Figure 4. DTM-derived morphometric feature maps of the training area.

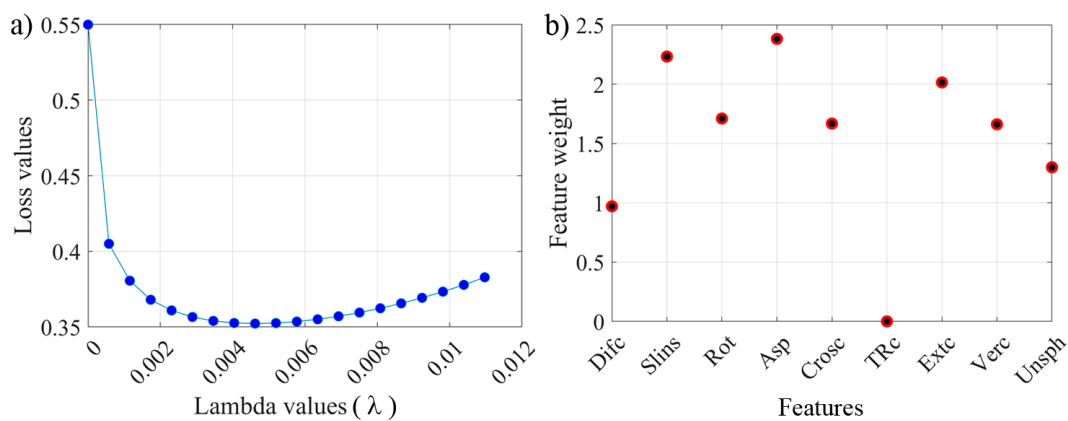


Figure 5. Feature selection using NCA. (a) Computing of optimized value of λ ; (b) Feature weights.

The significant features are those with a weight greater than the tolerance value as computed with Equation (2), i.e., greater than $T = 0.05$.

The chart in Figure 5b shows that only the TRc feature (Total Ring curvature) has a weight lower than the tolerance, it is even zero; TRc is a product of horizontal excess and vertical excess curvatures.

In our case, i.e., the classification of the three different coastal sections, this feature does not seem to improve the accuracy of the model.

The results obtained (Figure 5b) are in line with those found in the relevant literature. In particular, the most significant features are Slins and Asp; the other features are derived from the curvature computation and are those that affect less of the predictive ability of the classifier [29,30].

4.3. Supervised Machine Learning Classification

At first, models have been trained using all the nine computed features. Then, all models have been trained again after removing the TRc feature, which had resulted non-significant after the NCA. The “best” classifier, from the point of view of accuracy, was the k-NN (average score of 70%), and among the various algorithms, the best results are obtained with the Weighted k-NN. Similar comments can be made when eight predictors were used.

The k-NN classifiers [59] result very well suited when dealing with large volumes of training data; it is a non-parametric method, i.e., it makes no hypothesis about the distribution of the data being analyzed; the structure of the model is determined by the data itself and this is quite convenient, because in the “real world”, most data do not obey pre-established patterns. As such, it aims to better fit the input data by resulting in a greater level of accuracy in the classification. Thus, this model type is widely used for the generation of landslide susceptibility maps, since landslide distributions do not usually fit into neat distributions [25].

This type of classifier is closely dependent on the value of k (number of nearest neighbors), on the method of computing the distance (distance metric) and on the method chosen to determine the weight, as for the Weighted k-NN. Hence, we decided to use techniques of “hyperparameter optimization”, to identify the optimal parameters and compare them with those chosen. The parameters obtained in some cases were the same as those used; in other cases, they did not result in an improvement in terms of classification accuracy. In detail, for the k-NN classifiers, the optimal k was found to be 10, except for the Fine k-NN and Coarse k-NN classifiers for which the default values were used (1 and 100, respectively). For Weighted k-NN, the parameters that provided higher accuracy were the Euclidean distance and the inverse of the square of the distance for the computation of the weights, in agreement with the results obtained by “hyperparameter optimization”.

Figure 6 shows the plot of the accuracy scores; in panel (a), the score for all nine features, in panel (b) for only eight features and in panel (c), the differences of values.

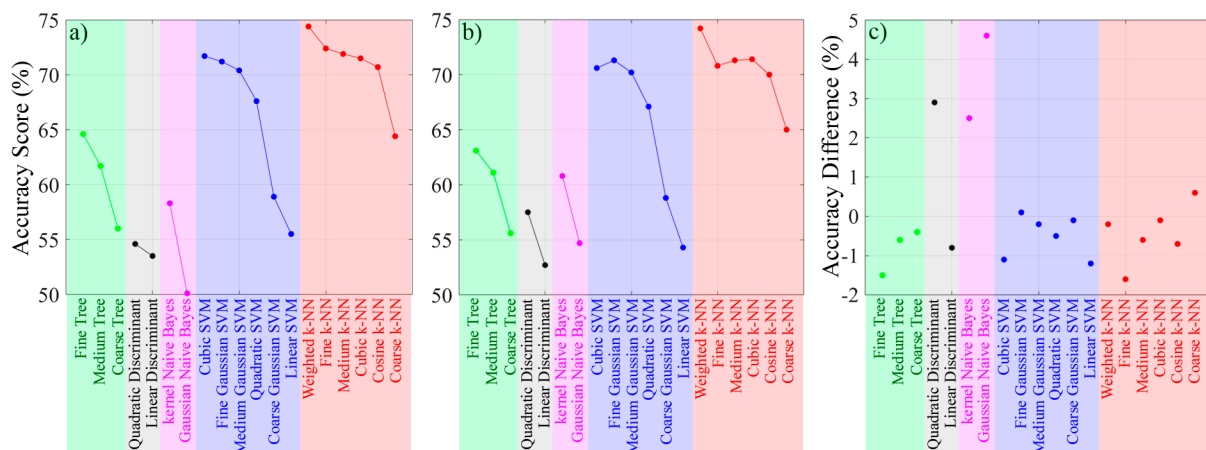


Figure 6. Accuracy score. (a) 9 features; (b) 8 features; (c) Accuracy score differences (9–8 features). The algorithms are sorted in ascending order of accuracy.

Within each of the five classification methods, the specific algorithm leads to significant differences in the level of accuracy up to 15 %.

Even looking at the differences between the scores (panel c), we noticed that removing the feature (TRc) did not result in significant changes in accuracy, except for a few algorithms (Quadratic Discriminant, Gaussian Naive Bayes, and Kernel Naive Bayes) for which there was a positive percentage change from 2.5 up to almost 5 points.

Looking also at the indices computed from the confusion matrices (*TPR*, *FNR*, *PPV*, *FDR*), shown in Figures 7 and 8, we noticed that removing the TRc feature resulted in a significant increase in Sensitivity (*TPR*) for Class II, thus decreasing the False Negative Rate (*FNR*). In addition, for those same algorithms, there was an increase in Precision (*PPV*) for all three classes. However, the associated accuracies are still too low, and thus they are not considered suitable to correctly represent the input data.

Models	Accuracy (%)	TPR (%)			FNR (%)			PPV (%)			FDR (%)		
		Class I	Class II	Class III	Class I	Class II	Class III	Class I	Class II	Class III	Class I	Class II	Class III
Fine Tree	64.6	73.1	62.2	61.3	26.9	37.8	38.7	68.7	63.4	62.8	31.3	36.6	37.2
Medium Tree	61.7	73.6	55.3	59.8	26.4	44.7	40.2	63.0	61.3	61.0	37.0	38.7	39.0
Coarse Tree	56.0	63.5	36.8	68.5	36.5	63.2	31.5	58.9	64.0	51.3	41.1	36.0	48.7
Linear Discriminant	53.5	64.5	48.5	50.9	35.5	51.5	49.1	60.4	54.4	48.3	39.6	45.6	51.7
Quadratic Discriminant	54.6	89.4	30.5	54.0	10.6	69.5	46.0	48.7	61.5	58.6	51.3	38.5	41.4
Gaussian Naive Bayes	50.1	86.6	29.3	45.5	13.4	70.7	54.5	44.1	59.9	54.0	55.9	40.1	46.0
Kernel Naive Bayes	58.3	77.8	49.3	53.9	22.2	50.7	46.1	57.9	59.8	57.4	42.1	40.2	42.6
Linear SVM	55.5	72.4	47.5	51.9	27.6	52.5	48.1	58.2	57.1	52.1	41.8	42.9	47.9
Quadratic SVM	67.6	79.6	62.7	64.2	20.4	37.3	35.8	72.0	65.2	66.4	28.0	34.8	33.6
Cubic SVM	71.7	84.0	71.2	64.2	16.0	28.8	35.8	77.3	67.0	72.3	22.7	33.0	27.7
Fine Gaussian SVM	71.2	73.3	60.5	79.7	26.7	39.5	20.3	86.5	73.7	63.2	13.5	26.3	36.8
Medium Gaussian SVM	70.4	80.3	68.9	65.3	19.7	31.1	34.7	78.1	65.1	70.3	21.9	34.9	29.7
Coarse Gaussian SVM	58.9	74.5	49.8	57.2	25.5	50.2	42.8	61.7	62.1	54.7	38.3	37.9	45.3
Fine k-NN	72.4	82.7	71.0	67.1	17.3	29.0	32.9	76.7	70.5	71.0	23.3	29.5	29.0
Medium k-NN	71.9	86.9	74.2	60.3	13.1	25.8	39.7	73.4	68.3	74.9	26.6	31.7	25.1
Coarse k-NN	64.4	86.9	60.5	53.6	13.1	39.5	46.4	61.5	64.1	68.1	38.5	35.9	31.9
Cosine k-NN	70.7	83.0	74.6	59.3	17.0	25.4	40.7	76.0	65.5	72.7	24.0	34.5	27.3
Cubic k-NN	71.5	87.3	74.4	58.7	12.7	25.6	41.3	72.7	68.0	74.6	27.3	32.0	25.4
Weighted k-NN	74.4	86.6	74.1	66.7	13.4	25.9	33.3	76.6	72.2	74.8	23.4	27.8	25.2

Figure 7. Model accuracy indicators trained with the 9 features (Difc, Slins, Rot, Asp, Crosc, TRc, Extc, Verc, Unsph); color brightness is directly proportional to percentage values.

Models	Accuracy (%)	TPR (%)			FNR (%)			PPV (%)			FDR (%)		
		Class I	Class II	Class III	Class I	Class II	Class III	Class I	Class II	Class III	Class I	Class II	Class III
Fine Tree	63.1	72.9	63.2	56.8	27.1	36.8	43.2	67.2	60.7	62.6	32.8	39.3	37.4
Medium Tree	61.1	74.0	55.7	57.6	26.0	44.3	42.4	61.3	61.3	60.6	38.7	38.7	39.4
Coarse Tree	55.6	63.2	31.1	72.8	36.8	68.9	27.2	61.1	68.3	49.9	39.9	31.7	50.1
Linear Discriminant	52.7	63.5	48.9	49.1	36.5	51.1	50.9	60.0	53.3	47.3	40.0	46.7	52.7
Quadratic Discriminant	57.5	89.4	41.0	52.0	10.6	59.0	48.0	50.9	63.2	62.4	49.1	36.8	37.6
Gaussian Naive Bayes	54.7	86.1	41.2	46.7	13.9	58.8	53.3	49.8	60.5	56.9	50.2	39.5	43.1
Kernel Naive Bayes	60.8	78.4	53.5	56.1	21.6	46.5	43.9	62.0	61.8	59.0	38.0	38.2	41.0
Linear SVM	54.3	72.7	48.0	48.3	27.3	52.0	51.7	58.1	54.9	50.7	41.9	45.1	49.3
Quadratic SVM	67.1	80.7	61.6	63.3	19.3	38.4	36.7	70.2	65.1	66.4	29.8	34.9	33.6
Cubic SVM	70.6	83.5	68.8	63.9	16.5	31.2	36.1	76.7	66.5	70.1	23.3	33.5	29.9
Fine Gaussian SVM	71.3	73.5	61.6	78.7	26.5	38.4	21.3	85.9	73.0	63.9	14.1	27.0	36.1
Medium Gaussian SVM	70.2	79.6	69.6	64.8	20.4	30.4	35.2	78.3	65.2	69.8	21.7	34.8	30.2
Coarse Gaussian SVM	58.8	74.7	48.8	57.7	25.3	51.2	42.3	61.6	62.4	54.4	38.4	37.6	45.6
Fine k-NN	70.8	81.4	69.3	65.3	18.6	30.7	34.7	76.0	68.7	69.1	24.0	31.3	30.9
Medium k-NN	71.3	86.1	74.1	59.3	13.9	25.9	40.7	72.8	67.6	74.4	27.2	32.4	25.6
Coarse k-NN	65.0	86.8	61.8	53.9	13.2	38.2	46.1	62.2	64.3	68.9	37.8	35.7	31.1
Cosine k-NN	70.0	82.5	73.8	58.6	17.5	26.2	41.4	76.1	64.9	71.2	23.9	35.1	28.8
Cubic k-NN	71.4	86.7	74.9	58.4	13.3	25.1	41.6	72.9	67.8	74.5	27.1	32.2	25.5
Weighted k-NN	74.2	85.8	74.0	66.9	14.2	26.0	33.1	77.4	71.5	74.4	22.6	28.5	25.6

Figure 8. Model accuracy indicators trained with the 8 features (Difc, Slins, Rot, Asp, Crosc Extc, Verc, Unsph); color brightness is directly proportional to percentage values.

The values of the computed indicators, as well as the accuracy scores, also showed that the Weighted k-NN algorithm is the one that provides higher accuracy than the others. For this algorithm, removing Feature TRc does not result in significant changes in sensitivity and accuracy values (differences of less than 2% for each class); the results are in line with those derived from the NCA method which showed TRc as a non-significant feature for our classification. Removing TRc also resulted in a slight balance of sensitivity and precision across classes.

The Weighted k-NN model is also characterized by an average value of flexibility (the propensity for the model to fit the data); this is significant because a less flexible model greatly reduces the chance of overfitting.

Optimization of the Weighted k-NN Classification

To evaluate the effectiveness of the NCA, the Weighted k-NN model was trained by also removing the features with a weight lower than 1.5, specifically the Difc feature with a weight of 1 and the Unsph feature with a weight of 1.4. The removal of Difc leads to an increase in overall accuracy of about 1% (accuracy score = 75%), whereas the removal of Unsph leads to a decrease in accuracy of about 2% (accuracy score = 73.3%).

The selection of the features to disregard followed an iterative procedure, according to the weights computed by the NCA method, is based on the evaluation of the accuracy achieved for training, in accordance with relevant literature [25]. This result highlights that the NCA played a key role in the choice of features; in our case, the accuracy of the model improves by removing only the feature with weights less than 1.

Figure 9 shows the confusion matrix and the four indicators derived from it; as for the final solution: Weighted k-NN trained after removing TRc and Difc features, having an accuracy score of 75%. Columns of the matrix represent the predicted values, whereas rows represent the actual values. The confusion matrix allows to deepen the analysis on the classification accuracy by providing a judgment on the correctness of the predictions; the classification error is recorded in the elements outside the main diagonal of the matrix. Panel (b) shows the Sensitivity values (*TPR*) and panel (c), the Precision values (*PPV*) for each class.



Figure 9. (a) Matrix of confusion; (b) *TPR* and *FNR*; (c) *PPV* and *FDR*.

From the graphs, it can be seen that a good balance has been reached between Sensitivity and Precision. The trained model shows high sensitivity mainly for Class I and Class II (between 74% and 87%) and good accuracy for all classes (greater than 72%). Models having high Sensitivity (*TPR*) have a higher probability to identify real cases, but they also have a rather high rate of false positives. High Precision Values (*PPV*), on the other hand, are indicators of confidence in having true positives and mean a reduction in false positives.

4.4. Validation of the Model Performance on the Training Area

The trained model was applied on the same training area using a distinct DTM. In detail, the DTM built on the test area derives from the same LiDAR data used for the training area, but the point cloud was first interpolated on a grid of 1 m step and then, by subsequent bilinear interpolation, a 5 m step DTM was built. So, the morphometric parameters obtained on the three coastal sections are slightly different from those used to train the model.

On the DTM of the test areas, the seven predictors (Slins, Rot, Asp, Crosc, Extc, Verc, Unsph) were computed and used in the trained model to obtain the class prediction vector of the results. The vector contains a number of class labels (I, II, III) equal to the number of observations given as input.

Figure 10 shows the map with 1 m contour lines, also derived from the DTM, overlaid on the output raster map containing the prediction.

The results shown in figures highlight that the model used produced a good classification in correspondence of the three coastal sections; the three areas are well delineated with the presence of a few isolated pixels of other classes. Class I (yellow pixels) in terms of morpho-evolutionary changes, corresponds to the initial and unperturbed phase, class II (green pixels) corresponds to the transition phase and class III (blue pixels) corresponds to active landslide processes.

The success rate of the model in the identification of the correct class in the learning areas could be better quantified through the Relative Operating Characteristic (ROC) curves that report for each class the True Positive Rate (*TPR*) as a function of False Positive Rate (*FPR*). Figure 11 shows the ROC curves for each class. The area under the curve (AUC) can assume values between 0.5 and 1. The greater the area under the curve (i.e., the closer the curve is to the top of the graph), the greater the discriminating power of the test and thus of the model. In detail, for Class I (Figure 11a), the test was highly accurate ($AUC > 0.9$), whereas for Classes II and III (Figure 11b,c), the test was moderately accurate ($0.7 < AUC < 0.9$). The judgments on the discriminative ability of the test, in terms of accuracy, were derived from the study by Swets [60].

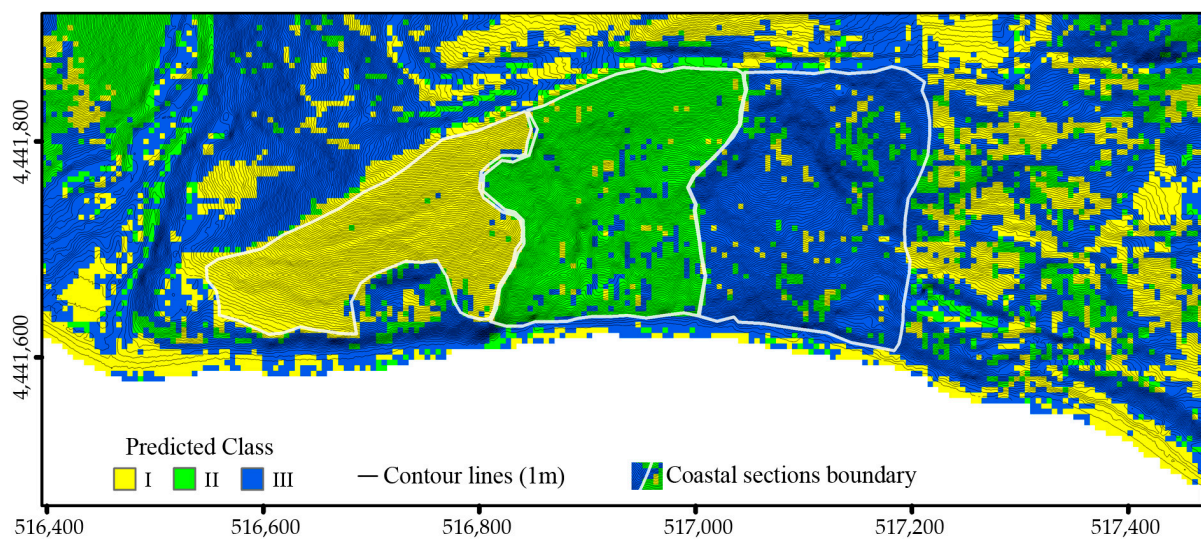


Figure 10. Classification overlaid on the test area DTM and the three coastal sections identified by expert judgment (white edge polygons).

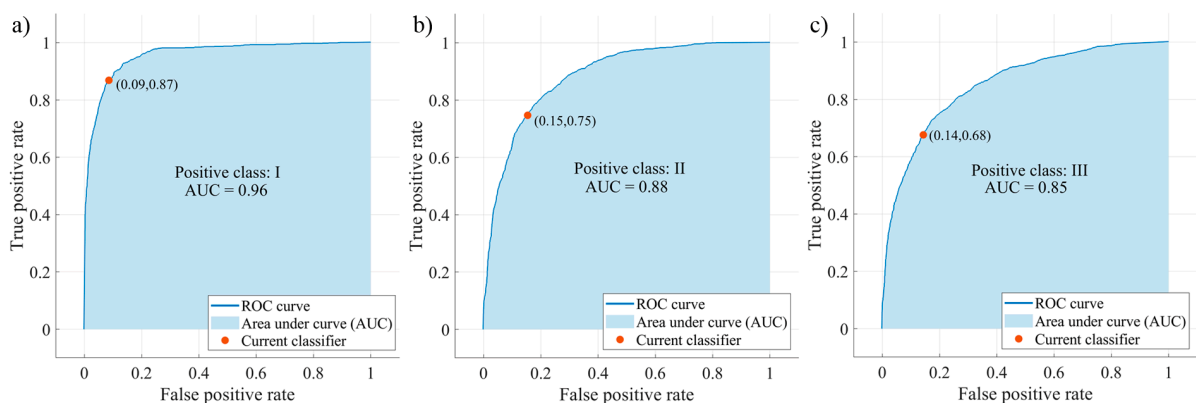


Figure 11. ROC curves. (a) Class I; (b) Class II; (c) Class III.

Looking at Figure 10, it shows that in the training area I (yellow), the AUC is 96% and there are very few pixels attributed to other classes, while for area III, where the AUC = 85%, there is a greater number of pixels of a different color from that of the class (blue).

With respect to the whole analyzed area, most of the areas classified as Class I are surrounded by Class II pixels; this is not accidental since the change from Class I (unperturbed area) to Class III (active landslide) implies, in most cases, the transitory phase represented by Class II.

The areas classified as Class II, coincide, in most cases, with the areas affected by gullies and ravines. The areas classified as Class III, in all cases, correspond to the areas characterized by active landslides (from inventory). The other areas not covered by landslides from inventory, however, correspond to areas characterized by a medium and high landslide hazard.

In details, the model was very effective for the classification of unperturbed areas (areas belonging to class I); this aspect is in line with the statistical results, which assign a highly accurate discriminating power to the class I predictive test (AUC = 0.96).

5. Discussion

To assess its applicability, the trained model was tested on the DTMs of two other coastal areas characterized by the same morpho-elevation phenomenon as the training area (Figure 2).

On these areas, the DTMs are also derived from the LiDAR data of the MATTM but only the points belonging to the bare ground surface have been triangulated. Next, the TIN was rasterized by linear interpolation to produce a grid DTM with 5 m resolution.

The seven predictors (Slins, Rot, Asp, Crosc, Extc, Verc, Unsph) selected to train the model were computed on the two DTMs.

Figure 12 (panel a) shows the result of the classification, superimposed on the contour lines with 5 m spacing, for the “Ripe Rosse” area. In the figure (panels b, c) are also shown two images (from Google Earth) of some details showing the presence of gullies and ravines; these shapes have been recognized and classified as part of class II, corresponding to the transition zone, affected by erosive and flow processes triggered by shallow retrogressive landslides.

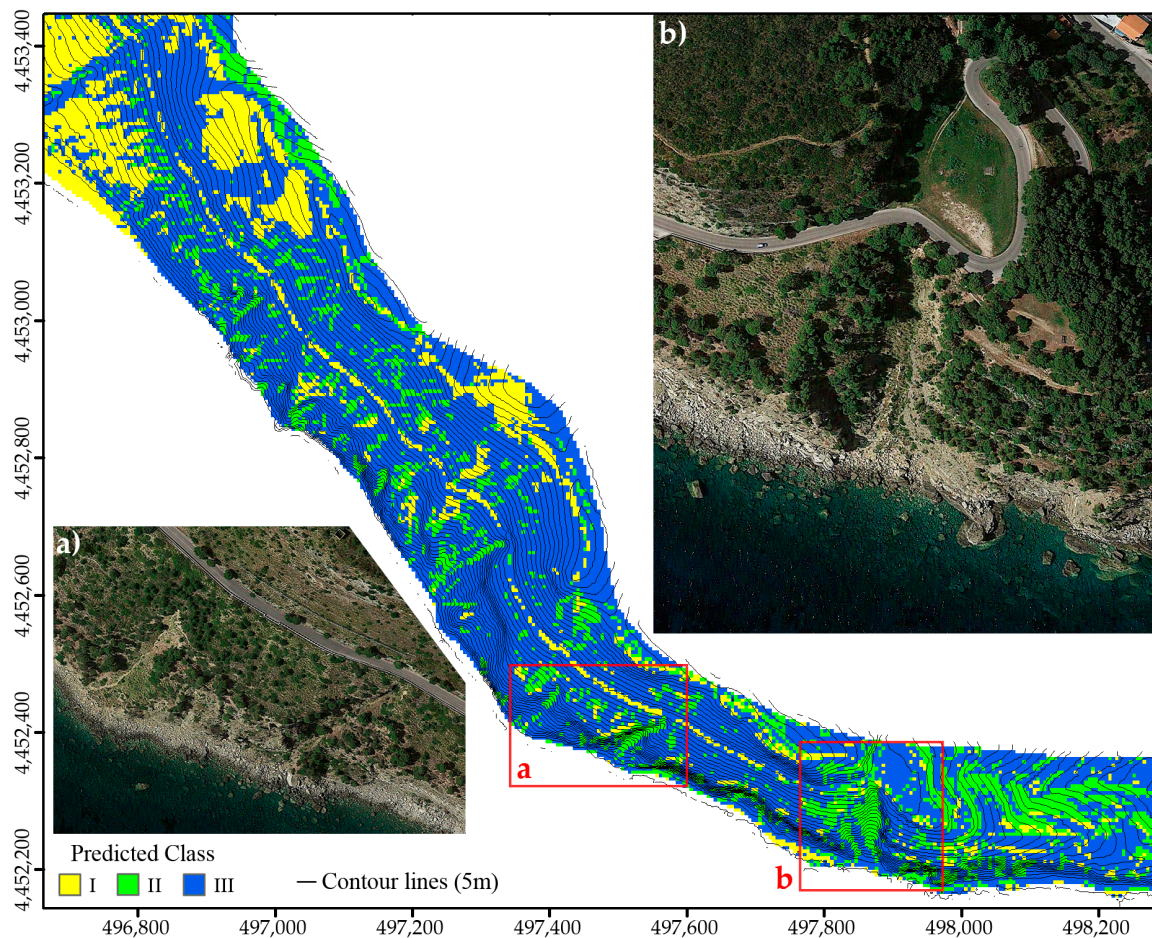


Figure 12. Test area “Ripe Rosse”. Classified map superimposed on contour lines; (a,b) details from Google Earth images.

The classification has produced well-defined clusters, highlighting very well the occurrence of channels (II) on the whole coastal landslide area (III). The area to the northwest, classified as Class I, is sandy beach, the only area not involved by landslides.

Figure 13 shows the classified map of the second area analyzed, in correspondence of the cliff of “Marina di Ascea”, superimposed on the contour lines. The figure shows that at the coastal zone (southern part), there are landslide areas with well distinguishable clusters belonging to class II. Figure 13a helps to better understand the morphology of the territory. Of particular interest are the clusters belonging to class I in correspondence of the coastal zone and in the central part (slightly eastward) of Figure 13b. In this area, on the satellite image, one can see a road descending towards the coast, not affected by landslide activities. Again, the trained model produced a classification with well-defined clusters, despite the morphological complexity of the test area.

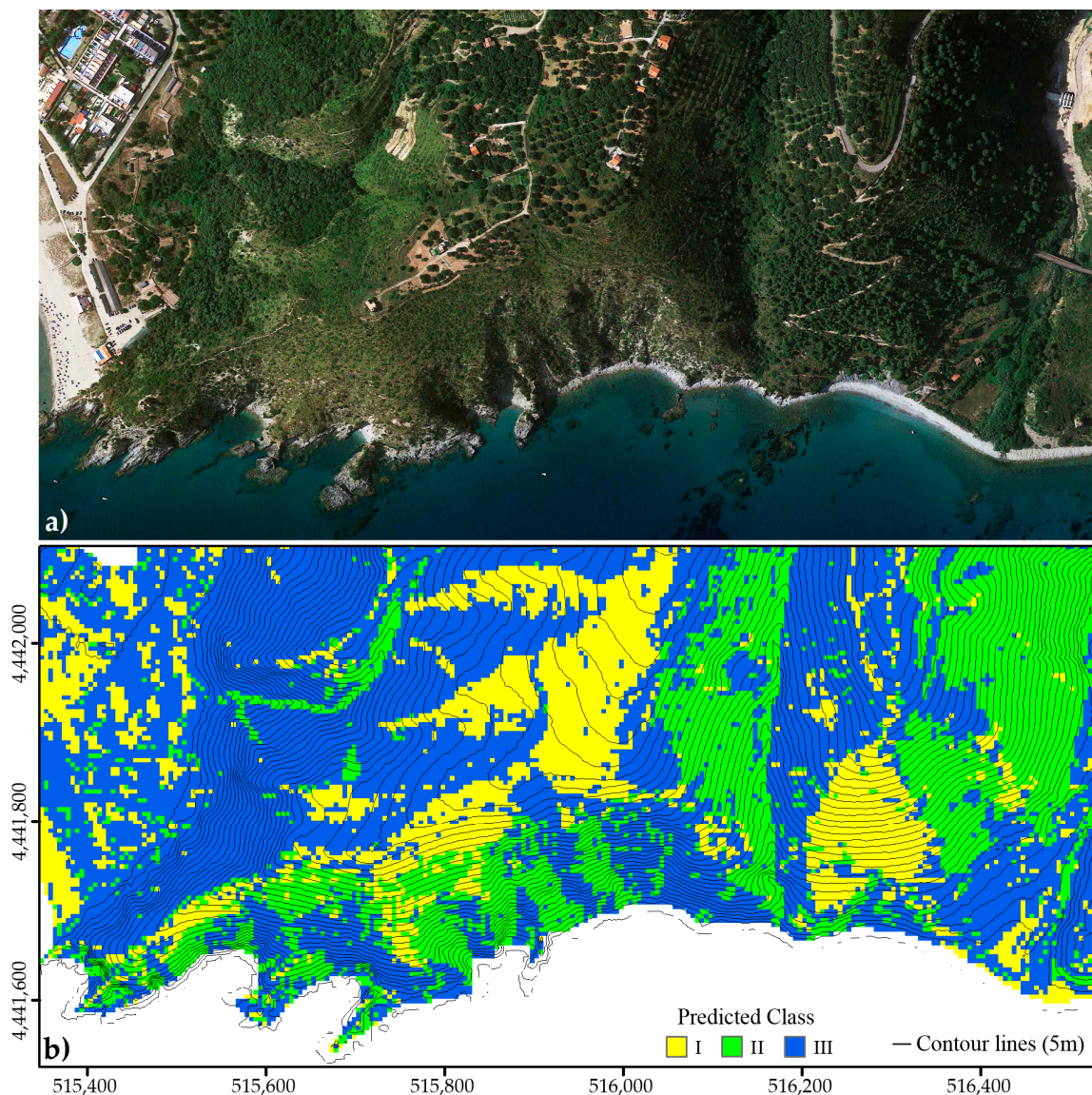


Figure 13. Marina di Ascea test area. (a) Image from Google Earth; (b) Classified map superimposed on contour lines.

The results are promising; they also highlight and confirm that predictive performance rates rely primarily on the quality of input data [61]. Most landslide inventories are incomplete and inaccurate [8], so it is important to have available a reliable and accurate landslide dataset to produce landslide susceptibility or classification maps [62]. The quality of the available LiDAR dataset is very high; in addition to plano-altimetric accuracy, the very high resolution allows the description of land topography to be extremely faithful [4].

LiDAR-based DTMs allow a better description and delimitation of landslide bodies at a larger scale [32]; it is not a chance that on the test cases, the output areas have well aggregated pixels, a non-trivial aspect not to be underestimated, being a Pixel-Based Approach (PBA).

The other important issue involves the DTM interpolation and the computation of morphometric parameters; as for the trained model, the accuracy is directly proportional to the accuracy of the input features [63]. Significant levels of accuracy of the model are achieved only if a rigorous process of data filtering is applied upstream for the extraction of the bare ground surface [4,6], which does not over-smooth the terrain and lose any representative shapes of the various evolutionary processes or landslide bodies.

Thus, in addition to allowing faithful reconstruction of landslide bodies, LiDAR technology is extremely useful in areas rich in vegetation since the high-intensity laser beam can penetrate even very dense vegetation [35].

Based on the aforementioned thinking, we have studied a methodology based on data able to render the objectivity of the territory, which only LiDAR or photogrammetric data can provide, trying to optimize the morphometric parameters derived by applying polynomials with powers greater than the square and removing non-objective data or not always available ones (such as rainfall data or stratigraphy).

The approach used (ALM) is of PBA type, whose main limitation, compared to object-oriented approaches (OOA), is to obtain spotty classification maps [64]; vice versa, OOA is more site-specific and contains too many classification steps to be transferred easily to other regions [65]. Although the approach used is PBA, the careful creation and selection of features to be given as input to the classifier, has produced results not far from an OOA approach, i.e., well-defined and clustered areas.

The classified maps obtained on the test areas shall be compared with those obtained from an expert-based classification, so as to validate the numerical results obtained.

6. Conclusions

In this study, several different Supervised Machine Learning models were analyzed to identify and classify a morpho-evolutionary phenomenon that characterizes a large part of the Cilento coast (slope-over-wall). In details, three different coastal sections have been classified, each characterized by a particular morpho-evolutionary process that has changed its shape (Class I unperturbed, Class II transitional and Class III active landslide).

The advantages of using ML techniques for mapping and monitoring landslide events are many; the algorithms are able to handle large datasets and the results are very accurate, especially if based on methods that avoid overfitting, such as cross-validation.

One of the concepts of our work was to use in the training phase only geometric data, i.e., morphometric parameters obtained from a high-resolution DTM derived from remotely sensed data. The results obtained are very promising and prove the capability of classifying and mapping these landslide phenomena accurately over large areas and without geotechnical data, which in most cases can only be acquired in situ. It is also evident that the additional use of other data sources can only make the classification more robust; the authors will experiment with this approach in the near future.

As for computing of morphometric parameters, the tests run confirmed the scientific findings reported in relevant literature, i.e., that polynomials of order higher than the second are more suitable to model more complex geomorphological forms with greater accuracy, aiding classification processes.

The most frequently cited features in the literature were tested, among the many that can be derived from a DTM. In order to avoid overfitting, and to remove features with low or no useful information value in our application, NCA was applied. This analysis has been preferred to the PCA (Principal Component Analysis) that is implemented in MATLAB, because NCA is a supervised method, so it needs class labels to be applied, unlike the PCA.

Identifying the most suitable ML model for a case study is likely to be challenging, since the results obtained from model training do not depend solely on the input data, but also depend on the uncertainties associated with modeling landslide phenomena and the limitations that characterize each model. Among all the models analyzed, the one that produced a higher accuracy has been the Weighted k-NN (accuracy rate of 75%). In addition, the analysis of ROC curves showed a very good discriminating power mainly for class I ($AUC > 0.96$) and a quite good discriminating power for the other classes ($AUC > 0.85$).

The trained models can be used to make predictions on different areas; the most relevant aspect is that the classification will be associated with a parameter that quantifies

its accuracy for each class. In this way, to each class can be quantitatively assigned a judgment of reliability; this would need to be validated by an expert geomorphologist.

Given the complexity of the Italian territory from the geomorphological point of view, which makes it subject to a high hydrogeological risk, on the analyzed area, as well as on all coastal areas, periodical ALS measurement campaigns are planned by the national competent agency. The authors therefore propose to apply again the proposed approach to the new data, with the aim of characterizing the areas from a multi-temporal point of view.

Author Contributions: Conceptualization, M.B. and M.F.; methodology, A.D.B., M.B. and M.F.; software, A.D.B.; validation, M.B. and M.F.; formal analysis, A.D.B., M.B. and M.F.; investigation, A.D.B.; visualization, A.D.B.; supervision, M.B. and M.F.; project administration and funding acquisition, M.F.; data curation, A.D.B.; writing—original draft preparation, A.D.B., M.B. and M.F. All authors have read and agreed to the published version of the manuscript.

Funding: The research is supported by the Italian Ministry of Education, University and Research under the National Project “Extended resilience analysis of transport networks (EXTRA TN): Towards a simultaneously space, aerial and ground sensed infrastructure for risks prevention”, PRIN 2017, Prot. 20179BP4SM.

Data Availability Statement: LiDAR data can be requested to Ministero dell’Ambiente e della Tutela del Territorio e del Mare (MATM) URL: “<http://www.pcn.minambiente.it/mattm/>” accessed on 1 September 2021.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Brunsden, D. Mass movement; the research frontier and beyond: A geomorphological approach. *Geomorphology* **1993**, *7*, 85–128. [[CrossRef](#)]
2. Niethammer, U.; James, M.R.; Rothmund, S.; Travelletti, J.; Joswig, M. Uav-based remote sensing of the super-sauze landslide: Evaluation and results. *Eng. Geol.* **2012**, *128*, 2–11. [[CrossRef](#)]
3. Godone, D.; Allasia, P.; Borrelli, L.; Gullà, G. Uav and structure from motion approach to monitor the maierato landslide evolution. *Remote Sens.* **2020**, *12*, 1039. [[CrossRef](#)]
4. Jaboyedoff, M.; Oppikofer, T.; Abellán, A.; Derron, M.-H.; Loye, A.; Metzger, R.; Pedrazzini, A. Use of lidar in landslide investigations: A review. *Nat. Hazards* **2012**, *61*, 5–28. [[CrossRef](#)]
5. Barbarella, M.; Di Benedetto, A.; Fiani, M.; Guida, D.; Lugli, A. Use of dems derived from tls and hrsi data for landslide feature recognition. *ISPRS Int. J. Geo-Inf.* **2018**, *7*, 160. [[CrossRef](#)]
6. Barbarella, M.; Cuomo, A.; Di Benedetto, A.; Fiani, M.; Guida, D. Topographic base maps from remote sensing data for engineering geomorphological modelling: An application on coastal mediterranean landscape. *Geosciences* **2019**, *9*, 500. [[CrossRef](#)]
7. D’Aranno, P.J.V.; Di Benedetto, A.; Fiani, M.; Marsella, M.; Moriero, I.; Palenzuela Baena, J.A. An application of persistent scatterer interferometry (psi) technique for infrastructure monitoring. *Remote Sens.* **2021**, *13*, 1052. [[CrossRef](#)]
8. Korup, O.; Stolle, A. Landslide prediction from machine learning. *Geol. Today* **2014**, *30*, 26–33. [[CrossRef](#)]
9. Mohan, A.; Singh, A.K.; Kumar, B.; Dwivedi, R. Review on remote sensing methods for landslide detection using machine and deep learning. *Trans. Emerg. Telecommun. Technol.* **2020**, *32*, e3998. [[CrossRef](#)]
10. Huang, F.; Cao, Z.; Guo, J.; Jiang, S.-H.; Li, S.; Guo, Z. Comparisons of heuristic, general statistical and machine learning models for landslide susceptibility prediction and mapping. *CATENA* **2020**, *191*, 104580. [[CrossRef](#)]
11. Al-Najjar, H.A.H.; Pradhan, B. Spatial landslide susceptibility assessment using machine learning techniques assisted by additional data created with generative adversarial networks. *Geosci. Front.* **2021**, *12*, 625–637. [[CrossRef](#)]
12. Shahabi, H.; Khezri, S.; Ahmad, B.B.; Hashim, M. Landslide susceptibility mapping at central zab basin, iran: A comparison between analytical hierarchy process, frequency ratio and logistic regression models. *CATENA* **2014**, *115*, 55–70. [[CrossRef](#)]
13. Borrelli, L.; Ciurleo, M.; Gullà, G. Shallow landslide susceptibility assessment in granitic rocks using gis-based statistical methods: The contribution of the weathering grade map. *Landslides* **2018**, *15*, 1127–1142. [[CrossRef](#)]
14. Aditian, A.; Kubota, T.; Shinohara, Y. Comparison of gis-based landslide susceptibility models using frequency ratio, logistic regression, and artificial neural network in a tertiary region of ambon, indonesia. *Geomorphology* **2018**, *318*, 101–111. [[CrossRef](#)]
15. Dastranj, A.; Noor, H. Comparative evaluation of landslide susceptibility map using ahp and fuzzy methods. *J. RS GIS Nat. Resour.* **2021**, *12*, 15–18.
16. Arabameri, A.; Chandra Pal, S.; Rezaie, F.; Chakraborty, R.; Saha, A.; Blaschke, T.; Di Napoli, M.; Ghorbanzadeh, O.; Thi Ngo, P.T. Decision tree based ensemble machine learning approaches for landslide susceptibility mapping. *Geocarto Int.* **2021**, 1–28. [[CrossRef](#)]

17. Youssef, A.M.; Pourghasemi, H.R.; Pourtaghi, Z.S.; Al-Katheeri, M.M. Landslide susceptibility mapping using random forest, boosted regression tree, classification and regression tree, and general linear models and comparison of their performance at wadi tayyah basin, asir region, saudi arabia. *Landslides* **2016**, *13*, 839–856. [[CrossRef](#)]
18. Huang, F.; Yao, C.; Liu, W.; Li, Y.; Liu, X. Landslide susceptibility assessment in the nantian area of china: A comparison of frequency ratio model and support vector machine. *Geomat. Nat. Hazards Risk* **2018**, *9*, 919–938. [[CrossRef](#)]
19. Song, Y.; Gong, J.; Gao, S.; Wang, D.; Cui, T.; Li, Y.; Wei, B. Susceptibility assessment of earthquake-induced landslides using bayesian network: A case study in beichuan, china. *Comput. Geosci.* **2012**, *42*, 189–199. [[CrossRef](#)]
20. Vahidnia, M.H.; Alesheikh, A.A.; Alimohammadi, A.; Hosseinali, F. A gis-based neuro-fuzzy procedure for integrating knowledge and data in landslide susceptibility mapping. *Comput. Geosci.* **2010**, *36*, 1101–1114. [[CrossRef](#)]
21. Tsangaratos, P.; Ilia, I. Comparison of a logistic regression and naïve bayes classifier in landslide susceptibility assessments: The influence of models complexity and training dataset size. *CATENA* **2016**, *145*, 164–179. [[CrossRef](#)]
22. Chang, Z.; Du, Z.; Zhang, F.; Huang, F.; Chen, J.; Li, W.; Guo, Z. Landslide susceptibility prediction based on remote sensing images and gis: Comparisons of supervised and unsupervised machine learning models. *Remote Sens.* **2020**, *12*, 502. [[CrossRef](#)]
23. Marjanović, M.; Kovačević, M.; Bajat, B.; Voženílek, V. Landslide susceptibility assessment using svm machine learning algorithm. *Eng. Geol.* **2011**, *123*, 225–234. [[CrossRef](#)]
24. Rodriguez, J.D.; Perez, A.; Lozano, J.A. Sensitivity analysis of k-fold cross validation in prediction error estimation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2010**, *32*, 569–575. [[CrossRef](#)]
25. Merghadi, A.; Yunus, A.P.; Dou, J.; Whiteley, J.; ThaiPham, B.; Bui, D.T.; Avtar, R.; Abderrahmane, B. Machine learning methods for landslide susceptibility studies: A comparative overview of algorithm performance. *Earth-Sci. Rev.* **2020**, *207*, 103225. [[CrossRef](#)]
26. Marc, O.; Hovius, N. Amalgamation in landslide maps: Effects and automatic detection. *Nat. Hazards Earth Syst. Sci.* **2015**, *15*, 723–733. [[CrossRef](#)]
27. Chen, X.; Chen, W. Gis-based landslide susceptibility assessment using optimized hybrid machine learning methods. *CATENA* **2021**, *196*, 104833. [[CrossRef](#)]
28. Youssef, A.M.; Pourghasemi, H.R. Landslide susceptibility mapping using machine learning algorithms and comparison of their performance at abha basin, asir region, saudi arabia. *Geosci. Front.* **2021**, *12*, 639–655. [[CrossRef](#)]
29. Novellino, A.; Cesarano, M.; Cappelletti, P.; Di Martire, D.; Di Napoli, M.; Ramondini, M.; Sowter, A.; Calcaterra, D. Slow-moving landslide risk assessment combining machine learning and insar techniques. *CATENA* **2021**, *203*, 105317. [[CrossRef](#)]
30. Nsengiyumva, J.B.; Valentino, R. Predicting landslide susceptibility and risks using gis-based machine learning simulations, case of upper nyabarongo catchment. *Geomat. Nat. Hazards Risk* **2020**, *11*, 1250–1277. [[CrossRef](#)]
31. Dou, J.; Yunus, A.P.; Merghadi, A.; Shirzadi, A.; Nguyen, H.; Hussain, Y.; Avtar, R.; Chen, Y.; Pham, B.T.; Yamagishi, H. Different sampling strategies for predicting landslide susceptibilities are deemed less consequential with deep learning. *Sci. Total. Environ.* **2020**, *720*, 137320. [[CrossRef](#)]
32. Pawłuszek, K.; Borkowski, A.; Tarolli, P. Towards the optimal pixel size of dem for automatic mapping of landslide areas. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2017**, *42*, 83–90. [[CrossRef](#)]
33. Hengl, T. Finding the right pixel size. *Comput. Geosci.* **2006**, *32*, 1283–1298. [[CrossRef](#)]
34. Pawłuszek, K.; Borkowski, A.; Tarolli, P. Sensitivity analysis of automatic landslide mapping: Numerical experiments towards the best solution. *Landslides* **2018**, *15*, 1851–1865. [[CrossRef](#)]
35. Lin, M.-L.; Chen, T.-W.; Lin, C.-W.; Ho, D.-J.; Cheng, K.-P.; Yin, H.-Y.; Chen, M.-C. Detecting large-scale landslides using lidar data and aerial photos in the Namasha-Liuoguyue Area, Taiwan. *Remote Sens.* **2014**, *6*, 42–63. [[CrossRef](#)]
36. Chen, W.; Li, X.; Wang, Y.; Chen, G.; Liu, S. Forested landslide detection using lidar data and the random forest algorithm: A case study of the three gorges, china. *Remote Sens. Environ.* **2014**, *152*, 291–301. [[CrossRef](#)]
37. Mezaal, M.R.; Pradhan, B.; Sameen, M.I.; Mohd Shafri, H.Z.; Yusoff, Z.M. Optimized neural architecture for automatic landslide detection from high-resolution airborne laser scanning data. *Appl. Sci.* **2017**, *7*, 730. [[CrossRef](#)]
38. Chen, T.; Trinder, J.C.; Niu, R. Object-oriented landslide mapping using zy-3 satellite imagery, random forest and mathematical morphology, for the three-gorges reservoir, china. *Remote Sens.* **2017**, *9*, 333. [[CrossRef](#)]
39. Li, X.; Cheng, X.; Chen, W.; Chen, G.; Liu, S. Identification of forested landslides using lidar data, object-based image analysis, and machine learning algorithms. *Remote Sens.* **2015**, *7*, 9705–9726. [[CrossRef](#)]
40. Kursu, M.B.; Rudnicki, W.R. Feature selection with the boruta package. *J. Stat. Softw.* **2010**, *36*, 1–13. [[CrossRef](#)]
41. Bergstra, J.; Yamins, D.; Cox, D. Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures. In Proceedings of the International Conference on Machine Learning PMLR, Atlanta, GA, USA, 17–19 June 2013; pp. 115–123.
42. Elith, J.; Leathwick, J.R. Species distribution models: Ecological explanation and prediction across space and time. *Annu. Rev. Ecol. Evol. Syst.* **2009**, *40*, 677–697. [[CrossRef](#)]
43. Pham, B.T.; Tien Bui, D.; Pourghasemi, H.R.; Indra, P.; Dholakia, M.B. Landslide susceptibility assessment in the uttarakhand area (india) using gis: A comparison study of prediction capability of naïve bayes, multilayer perceptron neural networks, and functional trees methods. *Theor. Appl. Climatol.* **2017**, *128*, 255–273. [[CrossRef](#)]

44. Chen, W.; Pourghasemi, H.R.; Panahi, M.; Kornejady, A.; Wang, J.; Xie, X.; Cao, S. Spatial prediction of landslide susceptibility using an adaptive neuro-fuzzy inference system combined with frequency ratio, generalized additive model, and support vector machine techniques. *Geomorphology* **2017**, *297*, 69–85. [[CrossRef](#)]
45. Guida, D.; Valente, A. Terrestrial and marine landforms along the cilito coastland (southern italy): A framework for landslide hazard assessment and environmental conservation. *Water* **2019**, *11*, 2618. [[CrossRef](#)]
46. Wang, H.; Zhang, L.; Yin, K.; Luo, H.; Li, J. Landslide identification using machine learning. *Geosci. Front.* **2021**, *12*, 351–364. [[CrossRef](#)]
47. Foroutan, M.; Kompanizare, M.; Ehsani, A.H. Semiautomatic morphometric land surface segmentation of an arid mountainous area using dem and self-organizing maps. *Arab. J. Geosci.* **2013**, *6*, 4795–4810. [[CrossRef](#)]
48. Carson, M.A.; Kirkby, M.J. *Hillslope Form and Process*; Cambridge University Press: London, UK, 1972.
49. Schmidt, J.; Evans, I.S.; Brinkmann, J. Comparison of polynomial models for land surface curvature calculation. *Int. J. Geogr. Inf. Sci.* **2003**, *17*, 797–814. [[CrossRef](#)]
50. Florinsky, I.V. Computation of the third-order partial derivatives from a digital elevation model. *Int. J. Geogr. Inf. Sci.* **2009**, *23*, 213–231. [[CrossRef](#)]
51. Evans, I.S. General geomorphometry, derivatives of altitude, and descriptive statistics. In *Spatial Analysis in Geomorphology*; Chorley, R.J., Ed.; Routledge: London, UK, 1972; pp. 17–90.
52. Goldberger, J.; Hinton, G.E.; Roweis, S.; Salakhutdinov, R.R. Neighbourhood components analysis. *Adv. Neural Inf. Process. Syst.* **2004**, *17*, 513–520.
53. Yang, W.; Wang, K.; Zuo, W. Neighborhood component feature selection for high-dimensional data. *JCP* **2012**, *7*, 161–168. [[CrossRef](#)]
54. Kotsiantis, S.B.; Zaharakis, I.; Pintelas, P. Supervised machine learning: A review of classification techniques. *Emerg. Artif. Intell. Appl. Comput. Eng.* **2007**, *160*, 3–24.
55. Zhao, X.; Yan, X.; Yu, A.; Van Hentenryck, P. Prediction and behavioral analysis of travel mode choice: A comparison of machine learning and logit models. *Travel Behav. Soc.* **2020**, *20*, 22–35. [[CrossRef](#)]
56. Sokolova, M.; Japkowicz, N.; Szpakowicz, S. Beyond accuracy, f-score and roc: A family of discriminant measures for performance evaluation. In Proceedings of the Australasian Joint Conference on Artificial Intelligence, Berlin, Heidelberg, Canberra, ACT, Australia, 29–30 November 2006; Sattar, A., Kang, B.-H., Eds.; Springer: Berlin/Heidelberg, Germany, 2006; pp. 1015–1021.
57. Ting, K.M. Confusion matrix. In *Encyclopedia of Machine Learning*; Sammut, C., Webb, G.I., Eds.; Springer: Boston, MA, USA, 2010; p. 209.
58. Hoo, Z.H.; Candlish, J.; Teare, D. What is an roc curve? *Emerg. Med. J.* **2017**, *34*, 357. [[CrossRef](#)] [[PubMed](#)]
59. Guo, G.; Wang, H.; Bell, D.; Bi, Y.; Greer, K. Knn Model-Based Approach in Classification. In Proceedings of the On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE, Berlin, Heidelberg, Rhodes, Greece, 21–25 October 2003; Meersman, R., Tari, Z., Schmidt, D.C., Eds.; Springer: Berlin/Heidelberg, Germany, 2003; pp. 986–996.
60. Swets, J.A. Measuring the accuracy of diagnostic systems. *Science* **1988**, *240*, 1285. [[CrossRef](#)]
61. Reichenbach, P.; Rossi, M.; Malamud, B.D.; Mihir, M.; Guzzetti, F. A review of statistically-based landslide susceptibility models. *Earth-Sci. Rev.* **2018**, *180*, 60–91. [[CrossRef](#)]
62. Chang, K.-T.; Merghadi, A.; Yunus, A.P.; Pham, B.T.; Dou, J. Evaluating scale effects of topographic variables in landslide susceptibility models using gis-based machine learning techniques. *Sci. Rep.* **2019**, *9*, 12296. [[CrossRef](#)] [[PubMed](#)]
63. Gaidzik, K.; Ramírez-Herrera, M.T. The importance of input data on landslide susceptibility mapping. *Sci. Rep.* **2021**, *11*, 19334. [[CrossRef](#)]
64. Keyport, R.N.; Oommen, T.; Martha, T.R.; Sajinkumar, K.S.; Gierke, J.S. A comparative analysis of pixel- and object-based detection of landslides from very high-resolution images. *Int. J. Appl. Earth Obs. Geoinf.* **2018**, *64*, 1–11. [[CrossRef](#)]
65. Van Den Eeckhaut, M.; Kerle, N.; Poesen, J.; Hervás, J. Object-oriented identification of forested landslides with derivatives of single pulse lidar data. *Geomorphology* **2012**, *173–174*, 30–42. [[CrossRef](#)]