On The Prediction of Landslide Occurrences and Sizes via Hierarchical Neural Networks

Quinton Aguilera

UCL: University College London

Luigi Lombardo (✉ l.lombardo@utwente.nl)

University of Twente Faculty of Geo-Information Science and Earth Observation: Universiteit Twente
Faculteit Geo-Informatie Wetenschappen en Aardobservatie

Hakan Tanyas

ITC: Universiteit Twente Faculteit Geo-Informatie Wetenschappen en Aardobservatie

Aldo Lipani

UCL: University College London

Research Article

Keywords: Landslide susceptibility, Landslide Hazard, Neural Networks, Slope Units, Hierarchical models

Posted Date: January 21st, 2022

DOI: <https://doi.org/10.21203/rs.3.rs-1260650/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

On the prediction of landslide occurrences and sizes via Hierarchical Neural Networks

Quinton Aguilera¹, Luigi Lombardo^{2*}, Hakan Tanyas², Aldo Lipani¹

Abstract

For more than three decades, the scientific community that studies landslides through data-driven models has focused on estimating where landslides occur across a given landscape. This concept is widely known as landslide susceptibility. And, it has seen a vast improvement from old bivariate statistical techniques to modern deep learning routines. Despite all these advancements, no spatially-explicit data-driven model is currently capable of also predicting how large landslides may be once they trigger in a specific study area. In this work, we exploit a model architecture that has already found a number of applications in landslide susceptibility. Specifically, we opt for the use of Neural Network (NN). But, instead of focusing exclusively on where landslides may occur, we extend this paradigm to also spatially predict classes of landslide sizes. As a result, we keep the traditional binary classification paradigm but we make use of it to complement the susceptibility estimates with a crucial information for landslide hazard assessment. We will refer to this model as Hierarchical Neural Network (HNN) throughout the manuscript.

To test this analytical protocol, we use the Nepalese area where the Gorkha earthquake induced tens of thousands of landslides in 2014. The results we obtain are quite promising. The component of our HNN that estimates the susceptibility outperforms a binomial Generalized Linear Model (GLM) baseline we used as benchmark. We did this for a GLM represents the most common classifier in the landslide literature. Most importantly, our HNN also suitably performed across the entire procedure. As a result, the landslide-area-class prediction returned not just a single susceptibility map, as per tradition. But, it also produced several informative maps on the expected landslide size classes.

Our vision is for administrations to consult these suite of model outputs and maps to better assess the risk to local communities and infrastructure. And, to promote the diffusion of our HNN, we are sharing the data and codes in a github repository in the hope that we would stimulate others to replicate similar analyses.

Keywords: Landslide susceptibility; Landslide Hazard; Neural Networks; Slope Units; Hierarchical models.

¹University College London, Department of Civil, Environmental and Geomatic Engineering, London, WC1E 6DE, United Kingdom

²University of Twente, Faculty of Geo-Information Science and Earth Observation (ITC), PO Box 217, Enschede, AE 7500, Netherlands

1 Introduction

Landslides are capable of inflicting large-scale human casualties and economic losses (Petley, 2012), and as a result they are widely researched. Continuing geoscientific advances have established routines to estimate where landslides may occur. This concept is referred to as susceptibility (Reichenbach et al., 2018). The standard definition of susceptibility corresponds to a relative likelihood of landslide occurrence within a specific mapping unit, under the control of a set of predisposing factors (Fell et al., 2008). For over three decades, this definition has been addressed by implementing different practices or techniques that have largely evolved following recent technological advances. During the early 1970’ies and until the 1980’ies, drawing susceptible slopes on a map was the result of a combination of field surveys and manual geomorphological mapping (Verstappen, 1983). As GIS made its appearance, bivariate statistical models followed closely and they became commonplace (Naranjo et al., 1994) until they were superseded by their multivariate counterpart (Süzen and Doyuran, 2004). The latter, in the form of binomial Generalized Linear Models (GLM), has then become the standard and is still the most widely used technique in the landslide susceptibility literature (Budimir et al., 2015; Reichenbach et al., 2018). However, the hypothesis that landslide occurrences are linearly related to a number of predisposing factors is a strong assumption that may not be supported by evidence in reality. Because of this, in recent years, a number of models capable of estimating nonlinear relations have been proposed as a further improvement to the more simplistic GLM framework. Some of these have simply translated the GLM context to a more flexible Generalized Additive Model (GAM) structure (Goetz et al., 2011; Lombardo and Tanyas, 2020; Steger et al., 2021). Others have taken the route of machine learning. Among these, support vector machines (e.g., Marjanović et al., 2011), decision trees (e.g., Lombardo et al., 2015) and neural networks (e.g., Lee et al., 2004; James et al., 2021; Anderson-Bell et al., 2021; Schillaci et al., 2021) have gained the spotlight within the geoscientific community.

However, irrespective of the algorithmic architecture, all these models have kept answering the same scientific question over the decades. They have sought to estimate where landslides may occur, being blind though to how large landslides may become, once they trigger in a given location. Conversely, this problem is typically addressed via physically-based approaches (Li et al., 2012; Bout et al., 2018; Van den Bout et al., 2021), although their applications over large regions is commonly hindered by geotechnical data availability. On the contrary, data-driven approaches can by-pass this issue by using proxies instead of geotechnical parameters. And yet, only a single contribution currently exists where landslide sizes (or their aggregated measure per slope units to be precise) are statistically estimated over territories whose scale is not suited for physically-based modeling. This work, authored by Lombardo et al. (2021), makes use of a GAM framework, under the assumption that landslide sizes (measured as planimetric areas) behave over space according to a log-Gaussian probability distribution. However, this work missed to address a fundamental problem. In fact, the use of a Gaussian likelihood implies that the bulk of the distribution is well esti-

mated but the tails are inevitably misrepresented. As a result, large landslides pertaining to the right tail of the size distribution are underestimated, hence underestimating the hazard to which local communities and infrastructures would be exposed to.

In this work, we take a different route, while still trying to combine some essential elements in the research progress described above. Firstly, we opted for Neural Network as our reference model structure. Such a framework is particularly appealing thanks to the widespread support and use in Python ([Van Rossum and Drake Jr, 1995](#)). Its libraries make codes easily shareable and the resulting analyses easily reproducible. Secondly, we modeled both the traditional landslide susceptibility together with the landslide sizes. However, differently from [Lombardo et al. \(2021\)](#), here we introduce the structure of a multi-label classifier rather than regressing the continuous distribution of landslide planimetric areas. Specifically, we run a Hierarchical Neural Network, where the susceptibility element is linked to the landslide susceptibility, and a multi-label element returns the probability of a specific landslide size class.

We do so by testing our modeling protocol over the co-seismic landslides triggered by the Gorkha earthquake in 2014. The landslide inventory generated by [Roback et al. \(2018\)](#) was accessed at the repository built by [Schmitt et al. \(2017\)](#); [Tanyaş et al. \(2017\)](#). And, the mapping unit we chose corresponds to Slope Units ([Carrara, 1988](#)).

The manuscript is structured as follows: Section 2 provides a description of the area where landslides occurred in response to the Gorkha earthquake, this being followed by a description of the mapping unit and predictors we chose (Section 3). In Section 6 we explain the Neural Network structure and how we made use in this work. Sections 7 and 8 present and comment the results, respectively. And ultimately, Section 9 summarizes the novelty of the work we present and anticipates future extensions.

We stress here that to promote similar applications we have shared data and codes in a github repository accessible at [link here](#).

2 Study area and landslide inventory

On April 25th 2015, the Gorkha earthquake 7.8 M_W ([Elliott et al., 2016](#)) struck the area in the proximity of Kathmandu, Nepal (see Figure 1a,b). The resulting losses amounted to approximately 9000 victims which, together with the infrastructural damage, made this event the worst natural disaster in Nepal since 1934 ([Kargel et al., 2016](#)). The earthquake affected-region covered an area of about 500 by 200 kilometers. And, a large portion of this area, also suffered from widespread landslides (see Figure 1c), with 24915 co-seismic failures and a total of 86.6 km² of planimetric landslide area (mean = 0.003 km²; standard deviation = 1.74 km²; maximum = 1.72 km²). Figure 1c shows both the spatial distribution of the mapped landslides and the boundary of the surveyed area by [Roback et al. \(2017\)](#) to generate the coseismic inventory. Their inventory is currently considered one of the most accurate and complete ([Tanyaş and Lombardo, 2020](#)) among the available global coseismic

107 inventories ([Schmitt et al., 2017](#); [Tanyaş et al., 2017](#)).

Figure 1: Panel *a* show the large scale geographic context; panel *b* zooms in to highlight the ground motion induced by the Gorkha earthquake; panel *c* depicts the spatial distribution of the coseismic landslides, this being shown in terms of spatial densities; panel *d* zooms further in to provide an overview of the slope unit partition we generated and its consistency with the underlying aspect distribution.

3 Mapping unit

We partition the study area into Slope Units (SUs). These correspond to a refinement of the SUs already generated and presented in [Tanyaş et al. \(2019\)](#) – and further details on their computation will be provided below. We remind here that a slope unit corresponds to the space bounded between ridges and streamlines, of catchments definable across different scales. Therefore, they efficiently depict the morphodynamics of a landslide and they also provide a spatial partition for which a single unit is independent from the others in terms of initial failure mechanism ([Alvioli et al., 2020](#); [Amato et al., 2019](#)). We generated the SUs by using *r.slopeunits*, a software developed and made available by [Alvioli et al. \(2016\)](#). The software optimizes the SU generation by maximizing within-unit homogeneity and outside-unit heterogeneity of the slope exposition ([Alvioli et al., 2018](#)), in a deterministic framework where the output is mainly controlled by four parameters ([Alvioli et al., 2016](#); [Castro Camilo et al., 2017](#)):

1. circular variance (*cv*), the parameter which regulates how strictly *r.slopeunits* should define homogeneity. The possible values are bound between 0 and 1, one allows for small variance of the aspect whereas the other extreme allows for large aspect variance.
2. flow accumulation threshold (*FAtresh*), this parameter initializes the search and it typically represents the starting size of the spatial partition, from which *r.slopeunits* further dissects the landscape into smaller units.
3. minimum SU area (*areamin*), a parameter representing the lower bound *r.slopeunits* tries to optimize for the SU delineation.
4. cleaning SU area (*cleansize*), a parameter representing the size below which *r.slopeunits* optimizes a merging routine where small units are fused into larger adjacent ones.

Here we start from the *r.slopeunits* parameterization used in [Tanyaş et al. \(2019\)](#). But, as their work was based on a global scale, the SU they delineated were too coarse for a site specific assessment. Therefore, we slightly modified the required parameterization and set the *cv* to 0.4, *FAtresh* to 500,000 m², *areamin* to 80,000 m², and *cleansize* to 50,000 m². Figure 1d shows the details of the SUs we generated, superimposed to the aspect for clarity. Overall, their resulting characteristics feature an average SU area of 0.22 km² and a standard deviation of 0.21 km²; the equivalent size of these two statistical moments being already a clear indication of how rough the landscape is at the transition between lesser and greater Himalayas ([Burbank, 2005](#)).

4 Predictors

The predictors we use are reported in Table 1. There we report ten predictors, nine of which (all but SU_A) have been calculated at a much finer spatial resolution compared to the SUs.

But, as our model is expressed at the SU scale, we summarized at this level the distribution of each predictors, via their respective mean and standard deviation values. As a result, the model we build features 18 predictors rather than nine, with the addition of the SU_A .

Table 1: Predictors’ summary table. Each parameter reported in the table, with the exception of the SU area, is expressed in a dual form through the mean and standard deviation computed within each SU. For conciseness, we do not list both terms in the table, but in the text they will be denoted by the suffixes μ and σ , added to the acronyms reported here.

Variable	Acronym	Units	Reference
Slope Unit Area	SU_A	m^2	e.g., (Lombardo et al., 2020)
Euclidean distance to stream	DIST2STR	m	e.g., (Ercanoglu and Gokceoglu, 2004)
Slope steepness	SLOPE	deg	(Zevenbergen and Thorne, 1987)
Vector ruggedness measure	VRM	unitless	(Sappington et al., 2007)
Profile curvature	PRC	1/m	(Heerdegen and Beran, 1982)
Planar curvature	PLC	1/m	(Heerdegen and Beran, 1982)
Eastness	EAST	unitless	e.g., (Lombardo et al., 2018)
Northness	NORTH	unitless	e.g., (Lombardo et al., 2018)
Peak ground acceleration	PGA	g	(Worden and Wald, 2016)
Topographic relief	RELIEF	m	(Jasiewicz and Stepinski, 2013)

Overall, each of the predictor variables are normally distributed, so standardization (mean of 0 and a standard deviation of 1; Ali et al., 2014) took place to regulate the scales of each variable. We also tried a min-max normalization (see Al Shalabi et al., 2006), but the standardization produced better performance (the result of these preliminary tests are not reported here), resulting in the choice to standardize the predictors instead of normalizing them.

5 Preprocessing

There does not currently exist any standard for classifying landslide size in terms of planimetric area. One of the most important reason one might consider dividing the landslides into classes based on area is to then be able to investigate how certain predisposing factors affect each class, as this may shed light upon how small to large landslides are catalyzed. In order to develop a method to optimally divide the landslide area (aggregated as the sum of all landslide planimetric areas per SUs) into classes, we used the Fischer-Jenks algorithm to derive the class boundaries (see, Jenks, 1967). The Fischer-Jenks algorithm determines the optimal break points by choosing those that minimize each category’s variance, though it requires information on how many breaks are to be used. To determine the optimal number of breaks, various break counts were trialed in ascending order, starting with two, on each iteration measuring the Goodness of Variance Fit (GVF). GVF is a value between 0 and 1 and indicates how well the categories produced by the Fischer-Jenks algorithm reflect the “natural breaks” in the data (Khamis et al., 2018). This iterative process continues until the

number of breaks associated with a GVF value beyond a certain threshold is reached, in this case, the value chosen was 0.85. Class 0 landslides are SUs with an area of 0, indicating that a landslide did not occur. The remaining non-zero values were broken into 5 classes using the Fischer-Jenks method described above, resulting in the six classes described in Table 2.

Table 2: Preliminary Landslide Class Structure. Bounds are expressed in m^2 . Records denoted with * are the smallest observed non-zero area subtract $1e^{-6}$

Class Name	Minimum Bound (Not Inclusive)	Maximum Bound (Inclusive)	Frequency
Class 0 (No landslide)	0	0	119932
Class 1	0.054690*	2981.98645	6517
Class 2	2981.98645	10821.34832	768
Class 3	10821.34832	29509.81466	137
Class 4	29509.81466	82850.15392	14
Class 5	82850.15392	182966.8526	1

However, two observations arose at this stage: firstly, the data is heavily imbalanced. This is not an uncommon issue when dealing with classification problems, and there are several ways of remedying this issue. In this case, we have opted to apply oversampling, which uses existing datapoints to interpolate artificial values, assigned to artificial records that closely resemble existing ones. The process of generating artificial data points occurs until each class has the same number of records, albeit with a proportion of them being synthetic (Chawla et al., 2002).

The second issue, however, is that for oversampling to take place, a minimum number of existing datapoints are required in each category to allow for interpolation. As is seen in Table 2, class 5 only has one record – an insufficient number to apply oversampling to. We have then overcome this issue by combining class 5 with class 4. We have also merged Class 2 into class 1 to reduce the number of tunable parameters by 33% overall, thus decreasing training time substantially and increasing performance as a result (more information available in Section 6.2). This produced the final classes detailed in Table 3.

Table 3: Preliminary Landslide Class Structure. Bounds are expressed in m^2 . Records denoted with * are the smallest observed non-zero area subtract $1e^{-6}$

Class Name	Minimum Bound (Not Inclusive)	Maximum Bound (Inclusive)	Frequency
Class 0 (No landslide)	0	0	119932
Class 1	0.054690*	10821.34832	7285
Class 2	10821.34832	29509.81466	137
Class 3	29509.81466	182966.8526	15

The next step was to create landslide presence/absence binary values. It was done by making a copy of the landslide class data and setting all non-class 0 values to 1, and all class

0 values to 0 (as numerical data). The final preprocessing done on the dependent variables was to encode the landslide class data so it could be used in a multi-class classifier model.

The result of these operations can be seen in Figure 2 where the panel *a* shows the traditional presence/absence landslide data, whereas panel *b* provides further insight into the classification scheme we adopted to run landslide-size-oriented prediction models.

Figure 2: Upper panel highlights the distribution of SUs labeled as stable or unstable; the lower panel provides additional graphical information on the classification scheme we opted for to predict specific ranges of landslide size classes, aggregated per SU.

5.1 Exploratory Data Analysis

Figure 2a shows that the spatial distribution of landslide occurrence does not appear to be uniform across the study area. Conversely, the overwhelming majority of SUs with landslides occurred in the Northern-Central sector. Small clusters of landslides also appear on the Southern border, although they are generally fewer in number, and even fewer are present in the Eastern and Western peripheries. Table 4 shows that the pattern described above results in a significant global spatial autocorrelation. This implies that there are spatial processes that influence the SUs landslide susceptibility, and this observation can be used to inform the interpretation of results, as good results should also reflect this spatial autocorrelation, likely induced by local amplification of the ground motion.

Table 4: Spatial Autocorrelation of Binary Landslide Occurrence.

Global Moran's I Spatial Autocorrelation

Moran's Index: 0.207678

Expected Index: -0.000008

p-value: 0.000000

Figure 2b shows the spatial distribution of landslides classes, using the categories defined in Section 5. Figure 2b further elucidates that the Northern sector features more slope units with landslide occurrences. And that as expected, the number of SU associated to Class 1 is predominant compared to Class 2, and that Class 2 is again more numerous than Class 3. This is a natural consequence of the landslide size distribution and it follows the the Frequency Area Distribution summarized in (Fan et al., 2019) for the Gorkha case.

6 Hierarchical Deep Neural Network for Landslide Class Prediction

The HNN implemented in this study is designed to predict the likelihood of landslide occurrence and size in spatial units, across the study area.

Hence, for each SU (see Section 3) and by using the variables introduced in Section 4 the model predicts the likelihood of landslide occurrence, and if a landslide is likely to occur, it also estimates what landslide-size-class is expected.

We have chosen to build a two output hierarchical neural network to guide the model's training (inductively bias the model) to focus on the binary occurrence of landslides and subsequently the size of the landslide. As a result, the model becomes more accurate. Another merit of having both outputs is that it enables a more straightforward comparison to existing approaches. The output predicting the landslide susceptibility can be used to compare our HNN to other models that perform the same task. This is the best way of comparing the model to existing approaches because there is no widely used or accepted

model that predicts size and therefore it is not possible to compare the landslide-size-class prediction with an equivalent baseline.

6.1 Model architecture

We stress here that the model we implemented is a single one. But for simplicity, we will separately refer for the rest of the manuscript to the two outputs, as *binary* and *class*. The former represents the traditional notion of landslide susceptibility or the likelihood of landslide occurrence. The latter refers instead to the estimation of the likelihood of a landslide belonging to a specific size class, reported in Table 3.

Figure 3: Conceptual framework of the proposed model

The proposed neural network architecture has a sequence of layers starting with: an input layer. This is followed by a Multi-Layer Perceptron (MLP). The aim of this MLP is to encode the input and provide the first classification, the binary one. This first MLP is then followed by another MLP, which takes as input the binary prediction and the encoded input through the first MLP. This second MLP outputs the class prediction. For clarity, we have graphically translated the architecture described above in Figure 3.

The loss functions for the binary and class outputs are the binary cross entropy and the categorical cross entropy respectively (details in Appendix A). The overall loss of the model is calculated as the linear combination of both losses. The parameter of the linear combination weights the importance of one classification vs the other. This loss equation is formally defined as:

$$\mathcal{L}_{\text{tot}} = \alpha \cdot \mathcal{L}_{\text{BCE}} + (1 - \alpha) \cdot \mathcal{L}_{\text{CCE}}, \quad (1)$$

where $\alpha \in [0, 1]$ is an hyper-parameter.

Further information on the model architecture and activation functions are also provided in Appendix A.

6.2 Performance Metrics

Throughout the landslide literature, the most common form of evaluation metric for landslide susceptibility is the Area Under the Receiver Operating Characteristic (ROC) curve, or AUC (Reichenbach et al., 2018). For consistency, the AUC will be used as the evaluation metric to compare our *binary* model with respect to a binomial GLM baseline. This will also be valid for the evaluation of the *class* output, as we will compute a ROC curve for each class. Moreover, there are also additional formats of the AUC that average the performance across

classes and allow for a single value to reflect the performance of the model on all classes. In this study, we will use both the one-vs-one weighted average and one-vs-rest weighted and unweighted average AUC to evaluate the performance of the class output. The one-vs-one approach generates a ROC curve for each pair of classes, where one class is considered as the positive example and the other as the negative example. Then, it computes their AUC and average them weighting each value based on the total number of examples belonging to the class pair.

The one-vs-rest approach generates a ROC curve for each class, where one class is considered as the positive example and the rest of the classes as the negative examples. Then, it computes their AUC and average them. In the weighted version, the weights are computed based on the total number of examples belonging to the selected class. The weighted version of these metrics is important because it is less affected by the class imbalance present in the dataset. These type of metrics are slight modification to the common AUC metric already used in the geomorphological literature. We opted to use them to provide a more complete model performance description and further details on this suite of performance metrics is reported in the Appendix A.

6.3 Influence of the Binary and the Class Components

The performance assessment we opted for also includes a loss weight ratio test. The aim of loss weight ratio testing is to be able to better understand the impact that the presence of the first output has on the second, and vice versa, as well as to find the optimal combination of losses in a Neural Network.

By prescribing loss weights in a multi-output model (adjusting α), it is possible to specify the extent to which each output’s loss contributes to the overall loss of the model. During the process of loss weight ratio testing, 11 combinations of weights at equally spaced integer intervals are tested, with α going from 0 to 1 at steps of 0.1. More information on this test is provided in Appendix A.

6.4 Ablation Analysis

To understand the importance of each feature or set of features, we perform an ablation analysis. This essentially consists in removing predictors from the dataset and retrain the model. The difference observed when using and not using a given predictor can be used as a measure of importance or explanatory power. In our dataset, we have 19 predictors. Eighteen of these can be thought as nine distinguishable signals that we convey to the model in a dual form, by using their mean and standard deviation values per SU. We recall here that the area of SU does not have a dual representation (refer to Table 1). It can therefore also be said that there are 10 independent categories of predictors, each one reflecting distinct attributes of the slope units. When conducting the ablation study, rather than removing a single component of the dual signal, only to leave its counterpart in the dataset, we opted to

remove both. The results will be shown in Section 7.1.1 and 7.2.1, for the *binary* and *class* models, respectively.

7 Results

7.1 Binary benchmark

The first stage of our modeling protocol features a benchmark step where the *binary* component of our HNN is compared to a traditional binomial GLM (Atkinson and Massari, 1998; Lombardo and Mai, 2018), for the latter represent the most common modeling routine used to assess landslide susceptibility (Budimir et al., 2015; Reichenbach et al., 2018).

Figure 4: Panel *a* summarized the evolution of the *binary* model through the epochs. Panel *b* offers a performance comparison between our *binary* model based on NN and a binary baseline based on a binomial GLM

Figure 4a shows the tests we have run to select the best *binary* model. The abscissa reports the number of epochs we tested our model for, up to 500 iterations. Examining the loss measured across epochs, the minimum corresponds to the 27th epoch, which we selected as our best model. Figure 4b shows the difference in performance between the best model

explained before and the baseline. Specifically, our *binary* NN (AUC = 0.88) better performs compared to the binomial GLM (0.82) one, albeit both fall in the excellent performance class proposed by (Hosmer and Lemeshow, 2000). This slight difference is noticeable in Figure 5 where in the central portion of the study area, the co-seismic susceptibility is shown to have been estimated slightly higher in the baseline. However, the baseline also produces high susceptibility in the peripheries (e.g., north-western and south-eastern sectors) where no landslides occurred, this being the reason for the difference in performance among the two models.

Figure 5: Baseline (a) Vs NN (b) predictive models translated into map form. The two susceptibilities capture the overall pattern with the main differences being due to how smoothly or abruptly the probabilities transition to the adjacent SUs.

7.1.1 Binary ablation tests

Figure 6 provides an overview of the predictor importance for the *binary* model. This operation essentially corresponds to a Jackknife test (see, Lombardo et al., 2016; Shrestha and Kang, 2019), where one variable is removed at a time, measuring the drop in performance as the model loses supporting information. Therefore, important predictors induce a large performance drop. This is clearly the case for the PGA, whose removal implies a median decrease in performance of approximately 0.07 AUC, leading the model from outstanding to excellent performance. This is not surprising because the PGA represents the role of

ground shaking, which is the most important variable controlling the spatial distribution of earthquake-induced landslides (e.g., Lombardo et al., 2019; Tanyaş et al., 2019). As for the remaining predictors, they do not significantly differ one from the others for they induce performance drops similar in magnitude.

Figure 6: Ablation test run for the *binary* model. The AUC reported to the y-axis is measured each time a predictor variable is removed from the initial set. We stress here that the removal of a predictor implies taking away both the corresponding mean and standard deviation values per slope unit. The green line corresponds to the best *binary* model shown in Figure 4a, obtained at the 27th epoch.

7.2 Class HNN component

Our HNN concatenates the binary model reported above to the four landslide-size-classes. To assess whether this component performs well, similarly to the previous figure, we monitored the HNN model as it converged to its final form. We stress here that our HNN sequentially links the output of the binary case to the model component that probabilistically distinguishes the four landslide size classes. This is shown in Figure 7a, where the loss estimated for class model converged to the best solution at the 26th epoch. In panel 7b, this performance is further described in a separate form, one ROC curve for each of the classes under consideration. There, the effect of the sample size can be seen across the landslide-size-classes. Class 0 and class 1 returned smooth ROC curves as they are constructed by using large data samples. This becomes rougher already for Class 2, due to the reduced amount of landslide with an average dimension and it is further exacerbated for the ROC

curve of Class 3, where the frequency of landslide with extreme sizes is significantly less.

Figure 7: Panel *a* summarized the evolution of the *class* model through the epochs. Panel *b* offers a performance overview among the four classes.

Irrespective of the specific Class, our HNN suitably recognizes the four landslide size groups, with outstanding performances according to Hosmer and Lemeshow (2000).

The resulting susceptibility estimates are translated in map form in Figure 8. There, it is possible to appreciate the different information our model provides. The probability Class 0 map (not to have a landslide) is essentially the inverse of the probability Class 1 (to have a small landslide). This is expected because small landslides constitute the vast majority of the landslide size distribution. As a result, the two maps appear to depict inverse spatial patterns. As for the probability Class 2 and Class 3 maps, they provide the complementary information we sought when we devised this experiment. In fact, specific slope units are highlighted with probability of landslide size occurrence, and the number drastically decreases from one class to the other, as also empirically expected.

7.2.1 Class ablation tests

In analogy to the analyses run for the *binary* component of our HNN model, we featured a series of ablation tests also for the *class* model. These are shown in Figure 9, where the effect

Figure 8: Probability of landslide-class-size occurrence

of the predictors' removal is measured for the four landslide size classes separately. Similarly to what happened in the case of the *binary* ablation tests, for the Classes 0 and 1, taking away the PGA (both its mean and standard deviation per SU) induced the largest drop in performance among all the predictors. Interestingly, as the landslide size class increased, the drop induced by removing the PGA drastically reduced. For instance, in the case of Class 2 removing the PGA still induces the largest drop in performance among the predictors, but the drop is almost in line with removing any of the others. As for the Class 3, here PGA is not the most relevant predictor anymore. In fact, none of the predictors' removal cause a significant drop in the modelling performance. In other words, in Class 3, none of the predictors dominate the contribution of others. We believe this behavior to be linked to the fact, that extremely large landslides may not be only linked to the ground motion but they may rather initiate because of very localized landscape characteristics. For instance, structural features have long been recognized as factors that control the occurrence of large landslides (e.g., Chigira and Yagi, 2006; Peart, 1991; Tanyas et al., 2021). This is due to the fact that weakness planes could make hillslopes kinematically susceptible even under the conditions of strong material properties and/or low ground shaking (Goodman et al., 1989). This makes the identification of weakness planes quite important, although it mostly missing in regional scale landslide susceptibility assessment (Ling and Chigira, 2020) including this study as such surveys are practically not possible to carry out for large areas. This means that in the case of Class 3, none of the variables may replace the role of that missing component

individually. However, the high modelling performance also shows that the compound effect of all variables acts as a proxy to identify those mapping units associated with the largest landslides.

In fact, a closer look at the predictor’s removal effect shows that, for instance, VRM is as one of the variables responsible for the largest drop in performance. We recall here that the VRM expresses how rough the terrain is, this being linked in several studies to the rock mass strength (e.g., [Tanyaş et al., 2019](#)). In other words, when the roughness is high, the landscape is likely made of rock masses with higher strength and whenever the roughness is low, this may imply soft materials that drape over the landscape, accommodating the deformation of induced by local tectonic regimes in the form of hills rather than steep scarps. This type of considerations can be behind the role of VRM with respect to extremely large landslides as their failure initiation maybe linked to weakness planes such as fractures or fissures dissecting rocky outcrops. Conversely, a landscape characterized by low VRW values mostly correspond of gentle slopes, where the tectonic response likely gives rise to ductile rather than fragile deformations.

Figure 9: Ablation test run for each landslide size class estimated via the *class* model. For each class, the AUC reported to the y-axis is measured each time a predictor variable is removed from the initial set. We stress here that the removal of a predictor implies taking away both the corresponding mean and standard deviation values per slope unit. The horizontal lines correspond to the best *binary* model shown in Figure 7a, obtained at the 26th epoch.

7.3 Loss ratio consideration

To complete the description of the HNN we propose, we opted to share the results of additional tests we have run to assess how *binary* and *class* component relate to each other in terms of performances.

Unsurprisingly, as shown in Figure 10a, the AUC values produced when the outputs are turned off are substantially below those when it is turned on. For instance, in the case of the *binary* output, its AUC value was initially less than 0.5, meaning that it was worse than a random selection. The remainder of results are more similar and found in the 0.86 to 0.88 range. But, as we zoom in towards the upper bound of the AUC distribution, Figure 10b unveils an interesting behavior. The maximum performance of the binary output occurred when the weightings were 10 and 0 for binary and class outputs respectively. This means that the presence of the class output only hindered the performance of the binary output. More interestingly, this was not the case for the class output. The second-worst performance of the class output was when the binary output was turned off, better only than when the class output itself was turned off. In fact, as the class output’s weight decreases on subsequent iterations, its performance generally improves, reaching a maximum on the antepenultimate iteration of the eleven trials, when the weights are 8 and 2 for the binary and class outputs respectively.

This demonstrates that the presence of the binary output can be leveraged to improve the performance of the class output. The implications of this finding are that the model should be tuned differently depending on the desired output. When making predictions using the binary output alone, the class output should be turned off, whereas when making predictions with the class output, the setting should be considered much more carefully.

8 Discussion

The model we propose is the first of its kind in the literature for it combines the prediction of which slope are susceptible to fail in response to a ground motion disturbance, together with the prediction of which size-class the landslides are expected to fall into, per slope unit. When we originally thought of implementing this model, we envisioned the landslide size component to be modeled closely to what has been done in [Gallen et al. \(2017\)](#); [Lombardo et al. \(2021\)](#). Gallen and co-authors proposed the first model able to spatially predict landslide sizes via Newmark sliding block analysis, a physically-based model. Lombardo and co-authors proposed something analogous but framed in the context of statistical modeling. In this work we took inspiration from the second example mentioned above, where the landslide size was modeled by using the continuous representation of the planimetric areas transformed into the logarithmic scale. Here, we initially envisioned a Hierarchical Neural Network capable of jointly modeling landslide occurrences and sizes, further improving on previous research by featuring a hierarchical model and also by keeping the planimetric area in its original squared meter scale. With this idea in mind we have run a number of tests

Figure 10: a) generic overview of how the *binary* and *class* model interacted with each other; b) Details of the same overview in top AUC range.

but likely due to the heavy tailed distribution of the landslide sizes, our HNN struggled to reproduce the whole distribution. The *binary* component was never an issue, as also demonstrated by the number of successful ANN applications in the context of landslide modeling (Conforti et al., 2014; Amato et al., 2021a,b). As for the landslide size component, due to the long tailed shape of the landslide planimetric distribution, we opted for a solution with a dual advantage. We used a common solution to this type of problems, by slicing the distribution into four bins (Goel et al., 2010) and adopting a multi-label classifier to take on the task of probabilistically recognizing each bin. In addition to the fact that this type of solution is particularly common, we also exploited the added value brought by the Neural Network architecture itself. In fact, this type of hierarchical models are well studied within the artificial intelligence community, allowing us to bring years of already built-in experience to solve a problem that was still not addressed within the geomorphological community.

This structure produced outstanding results across the two model components, (*binary* and *class*). This is shown in Figures 4 and 7. And, such observation opens up the question whether this type of models should represent the next direction to be taken when modeling natural hazards. In fact, especially within the geomorphological community, the vast majority of applications are confined within the scope of predicting only landslide occurrences (i.e., susceptibility assessments). Much more rarely, the community extends its reach to assessing the hazard associated with a potential landslide occurrence. This happens because the definition of landslide hazard features a temporal and size components which are not easily modeled. The temporal element is rarely accounted for because until the appearance of semi-automated mapping routines, monitoring a given area and manually mapping multi-temporal inventories has been a very difficult and expensive task (see explanation in Guzzetti et al., 2012). As for the dimension of the landslides, for decades the community has been focused on estimating the landslide event magnitude (Malamud et al., 2004; Tanyaş et al., 2018), a single measure valid for a large population of landslides. Only recently, Lombardo et al. (2021) have proposed a model that spatially estimates how prone slopes may be to release landslides of specific sizes. The combination of the landslide susceptibility and size elements proposed in this work belong to a currently uncharted territory, and their use is not featured in any international guidelines. However, we believe that the type of information compressed in Figure 8 can be really important to support decision makers and generally for territorial management practices. Knowing or at least expecting which slope is more likely to release an extremely large landslide is an information that at least could save money when investing on slope stabilization. And at its best, it could help minimize the risk to human lives.

It is fair to point out that this type of models are at their infancy stage within the natural hazard community. Therefore, their potential success relies on further testing, before even considering making them part of any well established procedure. For this reason, we are sharing the data as well as the codes we wrote during this experiment. In such a way, we intend to promote repeatability and reproducibility of analogous experiments. For instance,

we have currently developed this tool in the framework of co-seismic landslides. But, it is still unclear whether the same HNN is applicable in case of event-based rainfall induced landslides. Or, further tests need to be run on multi-temporal inventories.

9 Concluding remarks

We present the first hierarchical data-driven model able to simultaneously estimate where landslides may occur and to which landslide size class they may belong. We believe this model to be particularly suited for landslide hazard assessment. We recognize that the proposed model does not yet yield a full landslide hazard assessment as it is blind to the expected temporal frequency of such landslides. It should also be recognized, however, that the proposed model provides significantly more information than existing landslide susceptibility models. What makes this model particularly appealing is that its architecture hierarchically links the *binary* and the *class* elements. In other words, all the results reported in this manuscript are produced via a single data driven model.

In addition to future tests of the very same model – which we hope to stimulate by sharing data and codes with the readers in a github repository –, we expect to venture into additional variations of the present scheme. For instance, the simplest iterations we envision simply reproduce similar tests on rainfall induced landslides or on geomorphological inventories that cover a large time span rather than being linked to a single event. And, we are already planning to extend the complexity even further. In fact, withing the very same hierarchical structure, we could also add the type of landslides in between the *binary* and the *class* components. This could provide a complete description of the landslide process, which would need to be “just” extended in time to fully solve any requirement reported in the standard landslide hazard definition.

10 Software and data availability

In the hope of promoting similar applications, we have shared data and codes in a github repository accessible at: [click here](#).

The original landslide inventory can also be downloaded from the global repository of earthquake-induced landslides, accessible at: [click here](#).

A Further details on the proposed model

A.1 Activation Functions

The binary output has a sigmoid activation function, which produces values in between 0 and 1. The sigmoid was chosen because it is a common output activation when making binary classifications (Ngah et al., 2016). The class output uses a SoftMax (Sharma and Sharma,

2017) activation function with four outputs, one for each possible classification of landslide size. SoftMax was also chosen based on its common usage for multiclass classification models (Banerjee et al., 2020; Kumar et al., 2020).

In the first batch of hidden layers, a Rectified Linear Unit (ReLU, Agarap, 2018) is used as the activation function for all layers, as is common in many contemporary neural networks (Varshney and Singh, 2021). In the second batch of hidden layers, Leaky ReLUs (Clevert et al., 2015) are used to combat the somewhat common dying ReLU (Lu et al., 2019) problem that was encountered when initially attempting to build the network with standard ReLUs. The dying ReLU problem occurs when a standard ReLU function learns a large negative bias term for its weights, and given the shape of the function, this can result in the function exclusively outputting zero. Using a Leaky Relu addresses this issue by allowing the function to generate a small negative output when the function returns a negative value (Maas et al., 2013). Using leaky ReLUs in the second batch of hidden layers allowed the network to successfully overcome the dying ReLU problem.

A.2 Regularization

On each neuron in the hidden and output layers, kernel regularization reduces the values of the weights learned to prevent overfitting to training data. The type of regularization used is L2. L2 regularization applies a penalty to the weights learned of each neuron and is proportional to the initial weights learned. This means that while the weights will decrease in magnitude and be “closer to zero”, they will never reach zero (provided that the initial weight is not zero) and this will prevent overfitting while maintaining the general distribution of the weights learned. The range of values tested for finding the optimal L2 value in hyperparameter tuning can be found in Table 5.

A.3 Dropout layers

Dropout approaches are technically considered a type of activity regularization but are discussed here separately because they operate in a way distinctly different from the L2 regularization. The key idea behind dropouts is to randomly remove neurons (and their connections) from a network during training. This is designed to prevent the neurons from coadapting excessively. Excessive co-adaption can lead to overtraining, and studies have shown that adding dropout layers to various kinds of neural network markedly reduces overfitting and increases performance (Park and Kwak, 2016). The neurons dropped in any given layer are chosen randomly, but the proportion of neurons that are dropped is defined by a predetermined dropout rate. In this study, the dropout rate is tested at values from 0.2 to 0.8 in increasing increments of 0.2. These values were chosen based on a study by Srivastava (2013), that found optimal dropout rates typically are found between these values.

This additional form of activity regularization is included because Srivastava (2013) find that coupling dropout rates with other forms of regularization can further improve the perfor-

mance of neural networks. In their study, they coupled dropout regularization with L2, and max-norm activity regularization, and find that in both instances, the coupled techniques outperformed any individual technique used on its own.

A.4 Loss functions

Binary cross entropy was used on the binary output, and categorical cross entropy on the class output. Binary cross entropy, also known as log-loss is defined by the formula below:

$$\text{Binary Cross Entropy} = - \sum_{i=1} y_i \cdot \log(\hat{y}_i) + (1 - y_i) \cdot \log(1 - \hat{y}_i) \quad (2)$$

where $\hat{y}_i$ is the predicted output value and y_i is the observed value of the i^{th} model output. This loss function penalizes the result based on the distance of the prediction from the observed value. This means that high loss values indicate larger distances of predictions from the observed values. Categorical cross entropy performs a very similar role but is formatted to deal with multiple classes (in this case four). The notation is the same as for binary cross entropy.

$$\text{Categorical Cross Entropy} = -\frac{1}{4} \sum_{i=1}^4 y_i \cdot \log(\hat{y}_i) \quad (3)$$

In this case, $\hat{y}_i$ is the probability an event will occur and the sum of all $\hat{y}_i$ values is equal to 1 (one of the events will occur). When used in this paper, the class 0 landslide is treated as an event even though it technically represents the absence of an event to ensure that the sum of all $\hat{y}_i$ probabilities is equal to one even if no landslide is likely. Finally, the total loss of the model is the sum of both output's loss:

$$\text{Overall Loss} = \text{Binary Cross Entropy} + \text{Categorical Cross Entropy} \quad (4)$$

A.5 Loss weight ratio testing

The aim of loss weight ratio testing is to be able to better understand the impact that the presence of the first output has on the second, and vice versa, as well as to find the optimal combination for producing results. During this stage, the loss equation 4 is updated to the equation below, where ω_1 and ω_2 are the weights for each of the individual loss functions.

$$\text{Overall Loss} = \omega_1 \cdot \text{Binary Cross Entropy} + \omega_2 \cdot \text{Categorical Cross Entropy} \quad (5)$$

By prescribing loss weights in a multi-output model, it is possible to specify the extent to which each output's loss contributes to the overall loss of the model. The objective is to examine how changes in these weights impact the learning process and results. As administered in this study, 11 combinations of weights are tested, with each combination of weights summing to 10, as shown in Figure XXX.

A.6 Optimizer

Recently introduced by [Kingma and Ba \(2014\)](#), the Adam optimizer was chosen based on its superior performance compared to other optimizers such as AdaGrad ([Lydia and Francis, 2019](#)) or RMSProp ([Wichrowska et al., 2017](#)). Adam is algorithm for gradient-descent optimization based around “adaptive estimates of lower order moments”. This means that as the algorithm iterates, the parameters change. In the case of gradient descent, this means the learning rate starts at a large rate and decreases as the process continues to ensure convergence occurs at the correct minimum. It is computationally efficient and well suited to problems that use large amounts of data and parameters ([Kingma and Ba, 2014](#)). The authors also suggest a learning rate of 0.001, although in this paper several learning rates are tested, visible in table 4. The keras default setting for the Adam optimizer (‘adam’) is also trialed.

A.7 Hyperparameter tuning

The range of tunable hyperparameters and the values tested is visible in Table 5:

Table 5: Table of tested hyperparameter values

Hyperparameter	Selection choices
Number of layers in the first batch of hidden layers	6, 12, 18, 24, 48
Number of layers in the second batch of hidden layers	6, 12, 18, 24
Number of neurons in each layer of the first batch of hidden layers	4, 8, 16, 32, 64
Number of neurons in each layer of the second batch of hidden layers	4, 8, 16, 32, 64
Learning rate of Adam optimizer	1e-2, 1e-3, 1e-4, “adam”
L2 penalty for kernel regularization applied to all hidden and output layers	1e ⁻³ , 1e ⁻⁴ , 1e ⁻⁵
Dropout rate	0.2, 0.4, 0.6, 0.8

Due to the size of the search space (there are over 100,000 combinations) and the number of records (120,000+) an exhaustive grid search would have been very computationally demanding. Therefore, a Montecarlo hyperparameter tuning approach was used instead.

Montecarlo tuning is a process that relies on repeated random sampling to determine appropriate hyperparameters ([Raychaudhuri, 2008](#)). In this case, the process consisted of generating a random combination of hyperparameters 1000 times, with non-duplicate combinations being added to a cache. The training data was then split into test and validation training sets at an 80:20 ratio and run for 300 epochs. An early stopping mechanism was implemented whereby if the validation loss did not decrease for 15 epochs, then the training process halted. Each set of parameters was tested 5 times and the average minimum

validation loss was added to the cache with its corresponding parameter combination. The parameter combination with the lowest average validation loss value was then selected as the final model.

References

- Agarap, A. F. (2018) Deep learning using rectified linear units (relu). arXiv preprint arXiv:1803.08375 .
- Al Shalabi, L., Shaaban, Z. and Kasasbeh, B. (2006) Data mining: A preprocessing engine. Journal of Computer Science **2**(9), 735–739.
- Ali, P. J. M., Faraj, R. H., Koya, E., Ali, P. J. M. and Faraj, R. H. (2014) Data normalization and standardization: a technical report. Mach Learn Tech Rep **1**(1), 1–6.
- Alvioli, M., Guzzetti, F. and Marchesini, I. (2020) Parameter-free delineation of slope units and terrain subdivision of Italy. Geomorphology p. 107124.
- Alvioli, M., Marchesini, I. and Guzzetti, F. (2018) Nation-wide, general-purpose delineation of geomorphological slope units in Italy. Technical report, PeerJ Preprints.
- Alvioli, M., Marchesini, I., Reichenbach, P., Rossi, M., Ardizzone, F., Fiorucci, F. and Guzzetti, F. (2016) Automatic delineation of geomorphological slope units with r.slopeunits v1.0 and their optimization for landslide susceptibility modeling. Geoscientific Model Development **9**(11), 3975–3991.
- Amato, G., Eisank, C., Castro-Camilo, D. and Lombardo, L. (2019) Accounting for covariate distributions in slope-unit-based landslide susceptibility models. a case study in the alpine environment. Engineering Geology **260**, In print.
- Amato, G., Fiorucci, M., Martino, S., Lombardo, L. and Palombi, L. (2021a) Earthquake-triggered landslide susceptibility in Italy by means of Artificial Neural Network .
- Amato, G., Palombi, L. and Raimondi, V. (2021b) Data-driven classification of landslide types at a national scale by using Artificial Neural Networks. International Journal of Applied Earth Observation and Geoinformation **104**, 102549.
- Anderson-Bell, J., Schillaci, C. and Lipani, A. (2021) Predicting non-residential building fire risk using geospatial information and convolutional neural networks. Remote Sensing Applications: Society and Environment **21**, 100470.
- Atkinson, P. M. and Massari, R. (1998) Generalised linear modelling of susceptibility to landsliding in the central Apennines, Italy. Computers & Geosciences **24**(4), 373–385.

- 611 Banerjee, K., Gupta, R. R., Vyas, K., Mishra, B. et al. (2020) Exploring Alternatives to
612 Softmax Function. arXiv preprint arXiv:2011.11538 .
- 613 Van den Bout, B., Lombardo, L., Chiyang, M., van Westen, C. and Jetten, V.
614 (2021) Physically-based catchment-scale prediction of slope failure volume and geometry.
615 Engineering Geology **284**, 105942.
- 616 Bout, B., Lombardo, L., van Westen, C. and Jetten, V. (2018) Integration of two-phase
617 solid fluid equations in a catchment model for flashfloods, debris flows and shallow slope
618 failures. Environmental Modelling & Software **105**, 1–16.
- 619 Budimir, M., Atkinson, P. and Lewis, H. (2015) A systematic review of landslide probability
620 mapping using logistic regression. Landslides **12**(3), 419–436.
- 621 Burbank, D. W. (2005) Cracking the Himalaya. Nature **434**(7036), 963–964.
- 622 Carrara, A. (1988) Drainage and divide networks derived from high-fidelity digital terrain
623 models. In Quantitative analysis of mineral and energy resources, pp. 581–597. Springer.
- 624 Castro Camilo, D., Lombardo, L., Mai, P., Dou, J. and Huser, R. (2017) Handling high pre-
625 dictor dimensionality in slope-unit-based landslide susceptibility models through LASSO-
626 penalized Generalized Linear Model. Environmental Modelling and Software **97**, 145–156.
- 627 Chawla, N. V., Bowyer, K. W., Hall, L. O. and Kegelmeyer, W. P. (2002) SMOTE: synthetic
628 minority over-sampling technique. Journal of artificial intelligence research **16**, 321–357.
- 629 Chigira, M. and Yagi, H. (2006) Geological and geomorphological characteristics of landslides
630 triggered by the 2004 mid niigata prefecture earthquake in japan. Engineering Geology
631 **82**(4), 202–221.
- 632 Clevert, D.-A., Unterthiner, T. and Hochreiter, S. (2015) Fast and accurate deep network
633 learning by exponential linear units (elus). arXiv preprint arXiv:1511.07289 .
- 634 Conforti, M., Pascale, S., Robustelli, G. and Sdao, F. (2014) Evaluation of prediction capa-
635 bility of the artificial neural networks for mapping landslide susceptibility in the Turbolo
636 River catchment (northern Calabria, Italy). Catena **113**, 236–250.
- 637 Elliott, J., Jolivet, R., González, P. J., Avouac, J.-P., Hollingsworth, J., Searle, M. and
638 Stevens, V. (2016) Himalayan megathrust geometry and relation to topography revealed
639 by the Gorkha earthquake. Nature Geoscience **9**(2), 174–180.
- 640 Ercanoglu, M. and Gokceoglu, C. (2004) Use of fuzzy relations to produce landslide sus-
641 ceptibility map of a landslide prone area (West Black Sea Region, Turkey). Engineering
642 Geology **75**(3-4), 229–250.

- Fan, X., Scaringi, G., Korup, O., West, A. J., van Westen, C. J., Tanyas, H., Hovius, N., Hales, T. C., Jibson, R. W., Allstadt, K. E. *et al.* (2019) Earthquake-Induced Chains of Geologic Hazards: Patterns, Mechanisms, and Impacts. Reviews of Geophysics .
- Fell, R., Corominas, J., Bonnard, C., Cascini, L., Leroi, E., Savage, W. Z. *et al.* (2008) Guidelines for landslide susceptibility, hazard and risk zoning for land-use planning. Engineering Geology **102**(3-4), 99–111.
- Gallen, S. F., Clark, M. K., Godt, J. W., Roback, K. and Niemi, N. A. (2017) Application and evaluation of a rapid response earthquake-triggered landslide model to the 25 April 2015 Mw 7.8 Gorkha earthquake, Nepal. Tectonophysics **714**, 173–187.
- Goel, S., Broder, A., Gabrilovich, E. and Pang, B. (2010) Anatomy of the long tail: ordinary people with extraordinary tastes. In Proceedings of the third ACM international conference on Web search and data mining, pp. 201–210.
- Goetz, J. N., Guthrie, R. H. and Brenning, A. (2011) Integrating physical and empirical landslide susceptibility models using generalized additive models. Geomorphology **129**(3-4), 376–386.
- Goodman, R. E. *et al.* (1989) Introduction to rock mechanics. Volume 2. Wiley New York.
- Guzzetti, F., Mondini, A. C., Cardinali, M., Fiorucci, F., Santangelo, M. and Chang, K.-T. (2012) Landslide inventory maps: New tools for an old problem. Earth-Science Reviews **112**(1-2), 42–66.
- Heerdegen, R. G. and Beran, M. A. (1982) Quantifying source areas through land surface curvature and shape. Journal of Hydrology **57**(3-4), 359–373.
- Hosmer, D. W. and Lemeshow, S. (2000) Applied Logistic Regression. Second edition. New York: Wiley.
- James, T., Schillaci, C. and Lipani, A. (2021) Convolutional neural networks for water segmentation using sentinel-2 red, green, blue (rgb) composites and derived spectral indices. International Journal of Remote Sensing **42**(14), 5338–5365.
- Jasiewicz, J. and Stepinski, T. F. (2013) Geomorphons—a pattern recognition approach to classification and mapping of landforms. Geomorphology **182**, 147–156.
- Jenks, G. F. (1967) The data model concept in statistical mapping. International yearbook of cartography **7**, 186–190.
- Kargel, J. S., Leonard, G. J., Shugar, D. H., Haritashya, U. K., Bevington, A., Fielding, E., Fujita, K., Geertsema, M., Miles, E., Steiner, J. *et al.* (2016) Geomorphic and geologic controls of geohazards induced by Nepal’s 2015 Gorkha earthquake. Science **351**(6269).

- Khamis, N., Sin, T. C. and Hock, G. C. (2018) Segmentation of residential customer load profile in peninsular Malaysia using Jenks natural breaks. In 2018 IEEE 7th International Conference on Power and Energy (PECon), pp. 128–131.
- Kingma, D. P. and Ba, J. (2014) Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 .
- Kumar, Y., Dubey, A. K., Arora, R. R. and Rocha, A. (2020) Multiclass classification of nutrients deficiency of apple using deep neural network. Neural Computing and Applications pp. 1–12.
- Lee, S., Ryu, J.-H., Won, J.-S. and Park, H.-J. (2004) Determination and application of the weights for landslide susceptibility mapping using an artificial neural network. Engineering Geology **71**(3-4), 289–302.
- Li, W. C., Li, H. J., Dai, F. and Lee, L. M. (2012) Discrete element modeling of a rainfall-induced flowslide. Engineering geology **149**, 22–34.
- Ling, S. and Chigira, M. (2020) Characteristics and triggers of earthquake-induced landslides of pyroclastic fall deposits: an example from Hachinohe during the 1968 M7. 9 Tokachi-Oki earthquake, Japan. Engineering Geology **264**, 105301.
- Lombardo, L., Bakka, H., Tanyas, H., van Westen, C., Mai, P. M. and Huser, R. (2019) Geostatistical modeling to capture seismic-shaking patterns from earthquake-induced landslides. Journal of Geophysical Research: Earth Surface **124**(7), 1958–1980.
- Lombardo, L., Cama, M., Conoscenti, C., Märker, M. and Rotigliano, E. (2015) Binary logistic regression versus stochastic gradient boosted decision trees in assessing landslide susceptibility for multiple-occurring landslide events: application to the 2009 storm event in Messina (Sicily, southern Italy). Natural Hazards **79**(3), 1621–1648.
- Lombardo, L., Fubelli, G., Amato, G. and Bonasera, M. (2016) Presence-only approach to assess landslide triggering-thickness susceptibility: a test for the Mili catchment (north-eastern Sicily, Italy). Natural Hazards **84**(1), 565–588.
- Lombardo, L. and Mai, P. M. (2018) Presenting logistic regression-based landslide susceptibility results. Engineering geology **244**, 14–24.
- Lombardo, L., Saia, S., Schillaci, C., Mai, P. M. and Huser, R. (2018) Modeling soil organic carbon with Quantile Regression: Dissecting predictors’ effects on carbon stocks. Geoderma **318**, 148–159.
- Lombardo, L. and Tanyas, H. (2020) Chrono-validation of near-real-time landslide susceptibility models via plug-in statistical simulations. Engineering Geology **278**, 105818.

- Lombardo, L., Tanyas, H., Huser, R., Guzzetti, F. and Castro-Camilo, D. (2021) Landslide size matters: A new data-driven, spatial prototype. Engineering Geology **293**, 106288.
- Lombardo, L., Tanyas, H. and Nicu, I. C. (2020) Spatial modeling of multi-hazard threat to cultural heritage sites. Engineering Geology p. 105776.
- Lu, L., Shin, Y., Su, Y. and Karniadakis, G. E. (2019) Dying relu and initialization: Theory and numerical examples. arXiv preprint arXiv:1903.06733 .
- Lydia, A. and Francis, S. (2019) Adagrad—an optimizer for stochastic gradient descent. Int. J. Inf. Comput. Sci **6**(5).
- Maas, A. L., Hannun, A. Y., Ng, A. Y. et al. (2013) Rectifier nonlinearities improve neural network acoustic models. In Proc. icml, volume 30, p. 3.
- Malamud, B. D., Turcotte, D. L., Guzzetti, F. and Reichenbach, P. (2004) Landslides, earthquakes, and erosion. Earth and Planetary Science Letters **229**(1-2), 45–59.
- Marjanović, M., Kovačević, M., Bajat, B. and Voženílek, V. (2011) Landslide susceptibility assessment using SVM machine learning algorithm. Engineering Geology **123**(3), 225–234.
- Naranjo, J. L., Van Westen, C. and Soeters, R. (1994) Evaluating the use of training areas in bivariate statistical landslide hazard analysis-a case study in Colombia. ITC journal (3), 292–300.
- Ngah, S., Bakar, R. A., Embong, A. and Razali, S. (2016) Two-steps implementation of sigmoid function for artificial neural network in field programmable gate array. ARPN journal of engineering and applied sciences **11**(7), 4882–4888.
- Park, S. and Kwak, N. (2016) Analysis on the dropout effect in convolutional neural networks. In Asian conference on computer vision, pp. 189–204.
- Peart, M. (1991) The kaiapit landslide: events and mechanisms. Quarterly Journal of Engineering Geology and Hydrogeology **24**(4), 399–411.
- Petley, D. (2012) Global patterns of loss of life from landslides. Geology **40**(10), 927–930.
- Raychaudhuri, S. (2008) Introduction to monte carlo simulation. In 2008 Winter simulation conference, pp. 91–100.
- Reichenbach, P., Rossi, M., Malamud, B. D., Mihir, M. and Guzzetti, F. (2018) A review of statistically-based landslide susceptibility models. Earth-Science Reviews **180**, 60–91.
- Roback, K., Clark, M., West, A., Zekkos, D., Li, G., Gallen, S. and Godt, J. (2017) Map data of landslides triggered by the 25 April 2015 Mw 7.8 Gorkha, Nepal earthquake. US Geological Survey data release .

- Roback, K., Clark, M. K., West, A. J., Zekkos, D., Li, G., Gallen, S. F., Chamlagain, D. and Godt, J. W. (2018) The size, distribution, and mobility of landslides caused by the 2015 Mw7.8 Gorkha earthquake, Nepal. Geomorphology **301**, 121–138.
- Sappington, J. M., Longshore, K. M. and Thompson, D. B. (2007) Quantifying landscape ruggedness for animal habitat analysis: a case study using bighorn sheep in the Mojave Desert. The Journal of wildlife management **71**(5), 1419–1426.
- Schillaci, C., Perego, A., Valkama, E., Märker, M., Saia, S., Veronesi, F., Lipani, A., Lombardo, L., Tadiello, T., Gamper, H. A., Tedone, L., Moss, C., Pareja-Serrano, E., Amato, G., Köhl, K., Dămăţircă, C., Cogato, A., Mzid, N., Eeswaran, R., Rabelo, M., Sperandio, G., Bosino, A., Bufalini, M., Tunçay, T., Ding, J., Fiorentini, M., Tiscornia, G., Conradt, S., Botta, M. and Acutis, M. (2021) New pedotransfer approaches to predict soil bulk density using wosis soil data and environmental covariates in mediterranean agro-ecosystems. Science of The Total Environment **780**, 146609.
- Schmitt, R. G., Tanyas, H., Jessee, M. A. N., Zhu, J., Biegel, K. M., Allstadt, K. E., Jibson, R. W., Thompson, E. M., van Westen, C. J., Sato, H. P., Wald, D. J., Godt, J. W., Gorum, T., Xu, C., Rathje, E. M. and Knudsen, K. L. (2017) An open repository of earthquake-triggered ground-failure inventories. U.S. Geological Survey Data Series 1064.
- Sharma, S. and Sharma, S. (2017) Activation functions in neural networks. Towards Data Science **6**(12), 310–316.
- Shrestha, S. and Kang, T.-S. (2019) Assessment of seismically-induced landslide susceptibility after the 2015 Gorkha earthquake, Nepal. Bulletin of Engineering Geology and the Environment **78**(3), 1829–1842.
- Srivastava, N. (2013) Improving neural networks with dropout. University of Toronto **182**(566), 7.
- Steger, S., Mair, V., Kofler, C., Pittore, M., Zebisch, M. and Schneiderbauer, S. (2021) Correlation does not imply geomorphic causation in data-driven landslide susceptibility modelling—Benefits of exploring landslide data collection effects. Science of the total environment **776**, 145935.
- Süzen, M. L. and Doyuran, V. (2004) A comparison of the GIS based landslide susceptibility assessment methods: multivariate versus bivariate. Environmental geology **45**(5), 665–679.
- Tanyaş, H., van Westen, C., Allstadt, K., Nowicki, A. J. M., Görüm, T., Jibson, R., Godt, J., Sato, H., Schmitt, R., Marc, O. and Hovius, N. (2017) Presentation and Analysis of a Worldwide Database of Earthquake-Induced Landslide Inventories. Journal of Geophysical Research: Earth Surface **122**(10), 1991–2015.

- 776 Tanyaş, H., Allstadt, K. E. and van Westen, C. J. (2018) An updated method for estimating
777 landslide-event magnitude. Earth surface processes and landforms **43**(9), 1836–1847.
- 778 Tanyaş, H., Hill, K., Mahoney, L., Fadel, I. and Lombardo, L. (2021) The world’s second-
779 largest, recorded landslide event: lessons learnt from the landslides triggered during and
780 after the 2018 Mw 7.5 Papua New Guinea earthquake .
- 781 Tanyaş, H. and Lombardo, L. (2020) Completeness Index for Earthquake-Induced Landslide
782 Inventories. Engineering geology **264**, 105331.
- 783 Tanyaş, H., Rossi, M., Alvioli, M., van Westen, C. J. and Marchesini, I. (2019) A global
784 slope unit-based method for the near real-time prediction of earthquake-induced landslides.
785 Geomorphology **327**, 126–146.
- 786 Van Rossum, G. and Drake Jr, F. L. (1995) Python reference manual. Centrum voor
787 Wiskunde en Informatica Amsterdam.
- 788 Varshney, M. and Singh, P. (2021) Optimizing nonlinear activation function for convolutional
789 neural networks. Signal, Image and Video Processing pp. 1–8.
- 790 Verstappen, H. T. (1983) Applied geomorphology: geomorphological surveys for
791 environmental development. Number 551.4 VER.
- 792 Wichrowska, O., Maheswaranathan, N., Hoffman, M. W., Colmenarejo, S. G., Denil, M.,
793 Freitas, N. and Sohl-Dickstein, J. (2017) Learned optimizers that scale and generalize. In
794 International Conference on Machine Learning, pp. 3751–3760.
- 795 Worden, C. and Wald, D. (2016) ShakeMap manual online: Technical manual, user’s guide,
796 and software guide. US Geol. Surv. .
- 797 Zevenbergen, L. W. and Thorne, C. R. (1987) Quantitative analysis of land surface topog-
798 raphy. Earth surface processes and landforms **12**(1), 47–56.