Machine Learning for Landslide Prediction in Northern End of Longmenshan Fault, Sichuan Province, China

Irvan Ramadhan (416183) | Ryan Bobby Andika (416609)

Abstract. The objective of machine learning study for landslides are to predict the occurrence of landslide events as well as to identify several parameters which influence the landslide susceptibility. The study area is in the Northern End of the Longmenshan Fault, Sichuan Province, China. In this study, the authors used both the IBM SPSS program and the python language program to analyze and model the data provided. In data preparation, several steps are performed including data cleaning, aspect transformation, lithology separation, factor of safety (fos) classification, transformation, and normalization. To test the model accuracy, data partition was applied. The main modeling technique performed with SPSS software including Decision Tree, Support Vector Machine (SVM), Neural Network and KNN with additional logistic regression, K-Means, and PCA which run in python. Models are evaluated by using the confusion matrix with precision, recall, f1 score and accuracy to help determine the most suitable model. In general, all models with an accuracy score of more than 0.8, could be used to analyzed landslide susceptibility. Based on the analysis of data provided; scarps distance, specific weight, and lithology of moraine play the major roles for landslide susceptibility.

Keywords: Machine Learning, Landslide Susceptibility, Confusion Matrix, Scarps Distance

1. INTRODUCTION

Knowledge Discovery in Databases (KDD), Data Mining, and Machine Learning are the terms that could not be separated from one to another when we faced a large amount of data. KDD is a process that consists of processing a huge volume of data to extract information or knowledge that can be reused by an expert in such domain or by a knowledge-based system to solve problems (Napoli, 2005). There are three major steps in KDD regarding Amedeo Napoli in his book in 2005: Data Preparation, Data Mining, and Interpretation of the extracted units. Data Mining itself is a process of identifying interesting and useful patterns from large databases in the sense of automation (Arentze, 2009). Therefore, it should be no confusion among these two terms as KDD tells us the big picture of how we analyze the data and Data Mining as one of the steps to recognize the pattern of the data. Moreover, Machine Learning is a study that uses computer algorithms to turn empirical data into useable models for extracting information (Edgar and Manz, 2017). By the algorithm of Machine Learning, we can conduct Data Mining. Associatively, Data Mining could support Machine Learning to teach its machine by supplying numerous data on it. In the end, all of these processes would be so useful to grasp knowledge from vast data that humans in common could be overwhelmed by it.

The objective of this study is we would like to gain our information of Landslide susceptibility from the data of Landslide with numerous features or variables that may drive such hazard in the Northern End of Longmenshan Fault, Sichuan Province, China. Landslide susceptibility is measure of the likelihood of Landslide occurred in an area from the basis of local Terrain Condition (Brab, 1984). Hence, specifically, we would like to know about what features or variables take a significant contribution to the Landslide event and produce its Machine Learning model to make a prediction of this hazard so that some preventive action could be declared in the future.

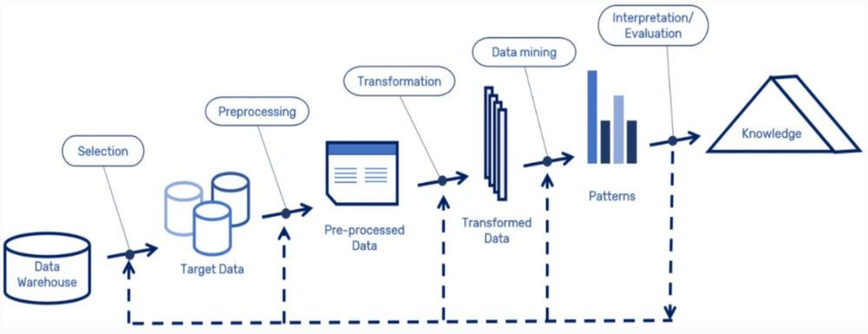


Figure 1. KDD Process Model (Rotondo and Quilligan, 2020)

2. STUDY SITE

In this research, we tried to analyze the observed data with various features from the severe earthquake on 12th May 2008, with magnitude 8.0, that shocked the Wenchuan area, Northwestern Sichuan Province, China, and led to the landslide-dammed lakes ("earthquake lakes"), debris flow, and rock avalanches - landslide which caused the most fatalities during that hazard event. The earthquake occurred along the Northeast-oriented Longmenshan fault zone due to the collision between the Indian Plate and Eurasian Plate in conjunction with the eastern part of the Tibetan block and western part of the Sichuan Basin (Cui et al., 2011). The Longmeshan Fault is composed of three branches of thrust faults (Back Fault, Central Fault, and Front

Fault), based on Cui et al. in 2011, and such hazard originated from the Center Fault after the seismic data investigation by China Earthquake Administration. The type of land failures was surveyed in three meizoseismal areas, Qingchuan county in Guangyuan city, Beichuan county in Mianyang city, and the epicenter area of Wenchuan county in Aba Tibetan Autonomous Prefecture (Gan and Zhang, 2019). From the field investigation, in the publication of Gan and Zhang in 2019, it was revealed that the typical lithology of the landslide deposit is the bedrock which consists of weak - strong and fine - coarse of granite, limestone, and sandstone, which was also covered by a large amount of loose clay and broken rock. The Longmen mount fault zone, and its surrounding area, have a humid subtropical climate which is common to the heavy rainfall (160 mm/day in 145 locations) between June and September, which is evident to the external dynamic conditions for the loose accumulation failure (Gan and Zhang, 2019). Therefore, by the geological fact, the area is prone to topographic movement like slide, collapse, erosion, and debris flow, even without the driving force of tectonic activity.

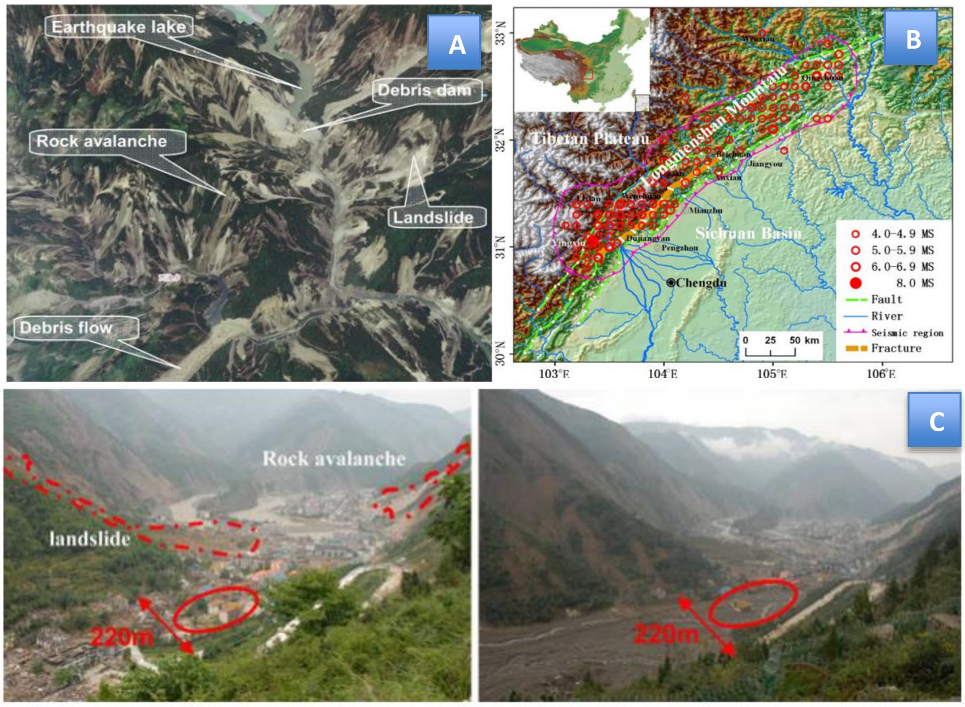


Figure 2. (**A**) A post-earthquake image of Tianchi village in the vicinity of Mianzhu City from Chinese State of Bureau of Surveying and Mapping. (**B**) Location of the Wenchuan earthquake and the aftershocks. (**C**) The destroyed scenes after earthquake combined with landslides (left) and after debris flow on September 24, 2008 (right) (Cui et al., 2011)

3. METHODS

3.1 General Data Processing Flowchart

The data processing flowchart for this **IBM SPSS** and **Python based project** consist of 4 major steps: Input Data (Blue), Data Preparation (Red), Machine Learning Model Creation and Evaluation (Brown).

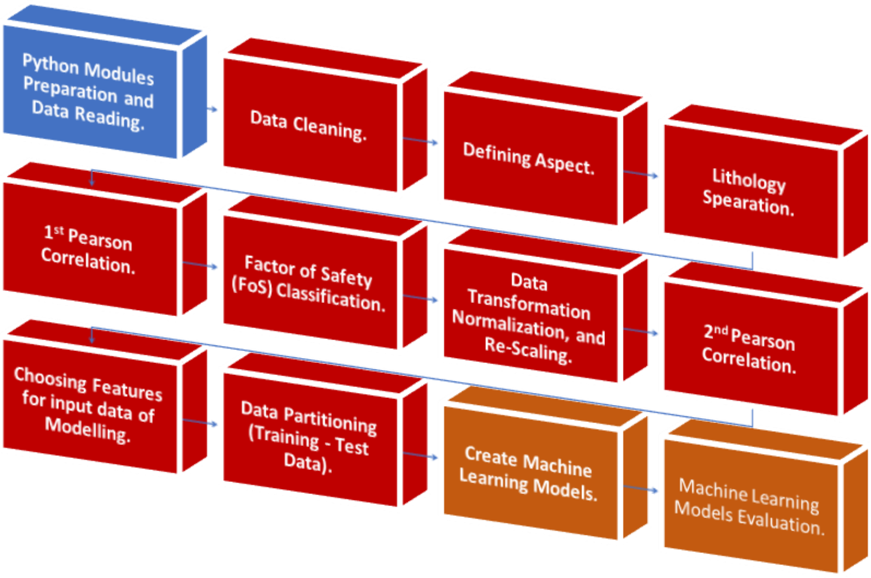

Python Modules Preparation and Data Reading.
Data Cleaning.
Defining Aspect.
Lithology Spearation.
1st Pearson Correlation.
Factor of Safety (FoS) Classification.
Data Transformation Normalization, and Re-Scaling.
2nd Pearson Correlation.
Choosing Features for input data of Modelling.
Data Partitioning (Training - Test Data).
Create Machine Learning Models.
Machine Learning Models Evaluation.

Figure 3. General Fata Procesiing Flowchart

3.2 Dataset

The dataset “LandslideData2021_26” consists of 21010 data points. Each data point contains 24 parameters which need to be analyzed to predict landslides hazard. Information about each individual parameters given in Table 1.

Table 1. Input parameters and their type as well as their role.

Parameters	Explanation	Type	Role
ID	ID of data point	Continuous	Record ID
ASPECT	Slope exposition [^{0}]	Continuous	Input
STRDIST	Distance to streams [m]	Continuous	Input
BASAREA	Catchment size [m^2]	Continuous	Input
BASIN	Catchment	Nominal	Input
CURVATURE	Slope curvature (+ convex; - concave)	Continuous	Input
CURVE_CONT	Contour-parallel slope curvature	Continuous	Input
CURVE_PROF	Profile-parallel slope curvature	Continuous	Input
CURVES	Change in height derived from slope [m]	Continuous	Input
DROP	Absolute change in height above valley floor [m]	Continuous	Input
ROCKDIST	Distance to rock [m]	Continuous	Input
FLOWDIR	Flow direction [^{0}]	Ordinal	Input
FOS	Factor of safety for shallow landslides	Continuous	Input
LITH	Rock type	Nominal	Input
ELEV	Elevation [m.a.s.l.]	Continuous	Input
COHESION	Cohesion [kN/m^2]	Continuous	Input
SLIDE	Landslide deposit yes/no	Flag	Target
SCARPDIST	Distance to failure scarps	Continuous	Input
SCARPS	Failure scarp yes/no	Flag	Input
FRICTANG	Friction angle [^{0}]	Continuous	Input
SLOPE	Slope gradient [^{0}]	Continuous	Input
SLOPELEG	Length of slope [m]	Continuous	Input
WOODS	Vegetation yes/no	Flag	Input
SPECWT	Specific weight [kN/m^3]	Continuous	Input

3.3 Theory

Model and algorithm in Machine Learning are divided mainly into supervised learning and Unsupervised Learning. In simple, Supervised Learning can be comprehended as a system trained to model a pattern or relationship in our already labeled data. However, Unsupervised Learning means that our system is introduced with unlabeled data so that computers should learn to model a pattern or relationship based on mathematic and statistics. As we already mentioned before, this project aims to create a Machine Learning system for Landslide Susceptibility. Therefore, several models and Machine Learning Techniques would be deployed in this project.

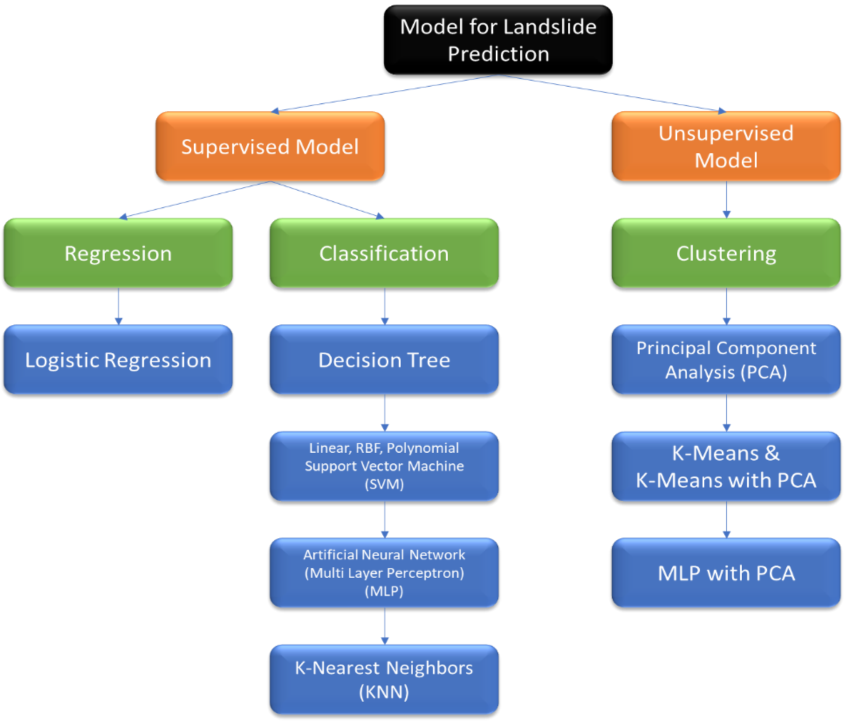

Model for Landslide Prediction
Supervised Model
Unsupervised Model
Regression
Classification
Clustering
Logistic Regression
Decision Tree
Principal Component Analysis (PCA)
Linear, RBF, Polynomial Support Vector Machine (SVM)
K-Means & K-Means with PCA
Artificial Neural Network (Multi Layer Perceptron) (MLP)
MLP with PCA
K-Nearest Neighbors (KNN)

Figure 4. Models for Machine Learning of Landslide Susceptibility

3.3.1 Supervised Machine Learning Model

a. Logistic Regression Model

Logistic regression is a robust and flexible method for dichotomous (binary) classification prediction; that is, it is used to predict for a binary outcome or state (Seufert, 2014). The logistic regression used a **Logistic Function** to create a model for binary data, in which have a **Sigmoid Curve function**.

$$\sigma(x) = \frac{1}{1+\exp(-x)} \quad (1)$$

$$f(p_n) = \log\left(\frac{\hat{p}_n}{1-\hat{p}_n}\right) = \hat{\beta}_0 + \hat{\beta}_0 x_{n1} + \cdots + \hat{\beta}_D x_{nD} \quad (2)$$

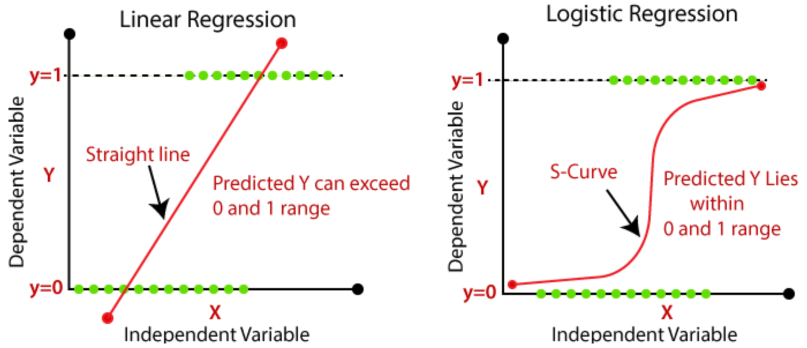


Figure 5. Linear Regression Function vs Logistic Regression Function (Jauregui, 2021)

b. Decision Tree Model

Decision tree is a technique in supervised machine learning that consists of internal nodes which related to attributes, edges which related to subsets of attribute or feature values, and terminal nodes (leaves) which related to class labels (Kononenko and Kukar, 2007). In Decision Tree, we should follow the **Ideal Tree Algorithm** which, in simple, we have to select a correct attribute as a node for the Decision Tree Splits. In order to get the correct attribute for a node, we could use the **Entropy** (H) or **Gini Impurity** (G) to measure the purity of split. We do the Decision Tree Splits by minimalize the total Entropy or Gini Impurity Value.

$$H = \sum_{i=1}^{N_i} -p(c_i)\log_2(p(c_i)) \quad (3)$$

$$G = 1 - \sum_{i=1}^{n} p^2(c_i) \quad (4)$$

Where $p(c_i)$ is the probability of class c_i in a node

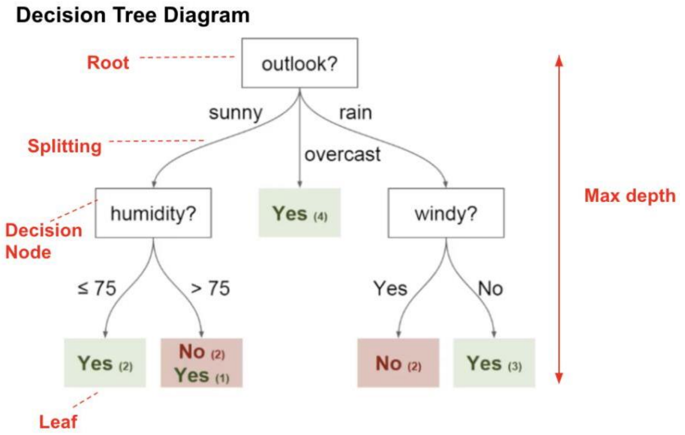


Figure 6. Decision Tree Model Visualization (programmersought.com, 2021)

c. Support Vector Machine (SVM) Model

Support vector machines (SVM) is a supervised machine learning method to analyze data and recognize patterns in which performs classification by creating an **N-dimensional hyperplane (a plane generalized into N dimensions)** that can optimally separate the data into two categories (Farber, 2012). However, the hyperplane could only give us a good result (maximize the Margin result) for a **Linearly separable dataset**. In **Non-Linearly separable data set**, which is common in many datasets, we could address it by **Soft Margin** or **Kernel Tricks** like Radial Basis Function (RBF), Polynomial, Sigmoid, and etc.

$$\hat{\beta}_0 + \hat{\beta}_0 x_1 + \cdots + \hat{\beta}_n x_n = 0 \qquad (5)$$

$$\emptyset(x, center) = \exp\left(-\gamma \|x = center\|^2\right) \qquad (6)$$

Where β is the model parameter for Hyperplane, x is data, γ is a Gamma constant, and $\emptyset$ is a new data feature. Actually, SVM (with Sigmoid Kernel) and Logistic Regression have a similar optimization problem. However, the Logistic Regression is more sensitive to outliers than SVM.

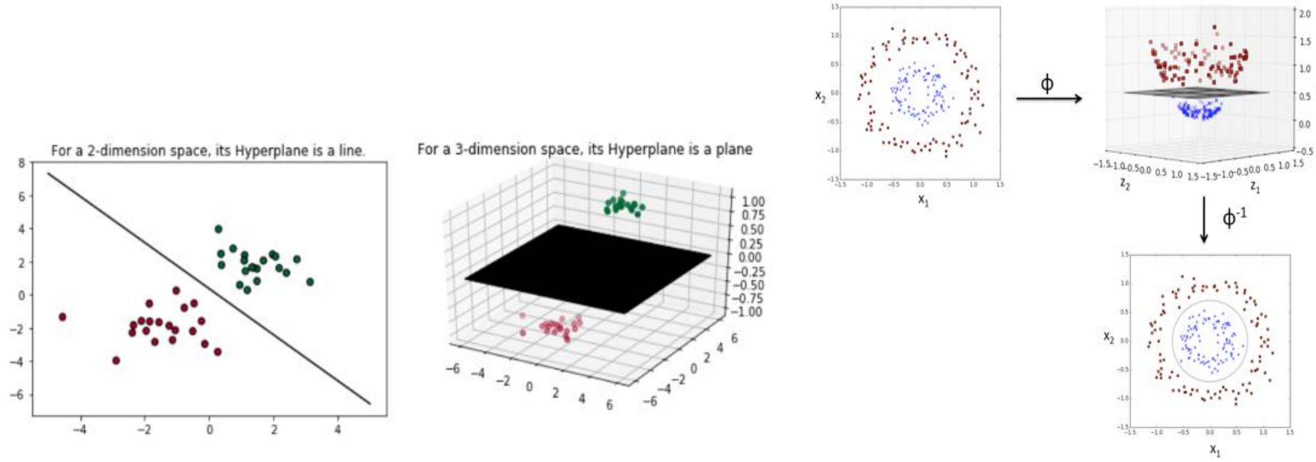


Figure 7. Linear Support Vector Machine Model Visualization Intuition (Chen, 2019)

c. K-Nearest Neighbor (KNN) Model

K-Nearest Neighbor is a one of the simplest Supervised Machine Learning model that implement the similarity of features to estimate a value of new data points in which such data points will be assigned a value based on the Eucledian Distance (d) it matched the points in the training set (Ma et al., 2020). Assume $\hat{p}$ and $\hat{q}$ are two points with its particular position, then the Euclidian Distance (d):

$$d(p, q)^2 = (q_1 - p_1)^2 + (q_2 - p_2)^2 \qquad (7)$$

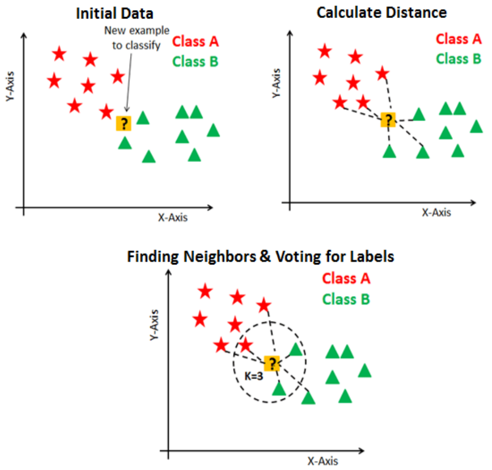


Figure 8. Illustration of the k-nearest neighbors (kNN) classifier (Bzdok, 2018)

3.3.2 Unsupervised Machine Learning Model:

a. K-Means Model

K-means clustering is an unsupervised model technique that reduces the data dimension by finding appropriate representatives or **Centroids** (center of each subset) for clusters or groups of data points (Subasi, 2020). In simple, the K-Means algorithm starts with the number of Centroids (or clusters) decision and deploys the first time Centroids location randomly. Then two iterative steps conducted until **threshold inertia** being reached, or being defined by the **maximum iteration**:

- Select the cluster for each sample in the sense of nearest position with centroid (in **Eucledian distance (d)**).
- Reposition centroid by averaging over all related samples.

In the end, the best result for K-Mean Models could be achieved when it reaches the overall lowest inertia.

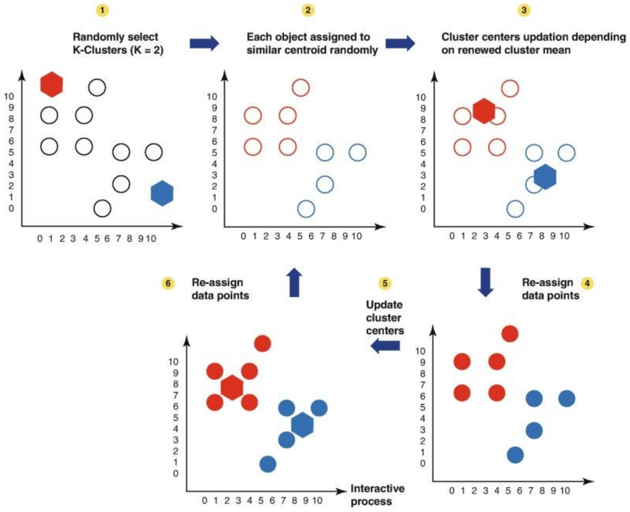

1
Randomly select K-Clusters (K = 2)
2
Each object assigned to similar centroid randomly
3
Cluster centers updation depending on renewed cluster mean
4
Re-assign data points
5
Update cluster centers
6
Re-assign data points
Interactive process
0 1 2 3 4 5 6 7 8 9 10

Figure 9. Sketch of the k-means clustering algorithm (Traverso et al., 2018)

b. Principal Component Analysis (PCA)

The inertia measurement assumes a cluster that **convex** and **isotropic**, which lead to the poorly K-Mean Clustering for elongated or irregular-shaped clusters (Wellmann and Chudalla, 2021). To overcome such a case, Principal Component Analysis could be a solution to a higher-dimensional case. **Principal component analysis (PCA)** is a multivariate **data dimensionality reduction technique**, used to simplify a data set to a smaller number of factors that explain most of the variability (variance) (Niculescu and Andrei, 2016). In the end, the result of PCA could be implemented in the hope of enhancing the accuracy of the model for any Machine Learning Methods. However, this enhancement still depends on the dataset itself, which sometimes also gives us no enhancement or even worsens the result.

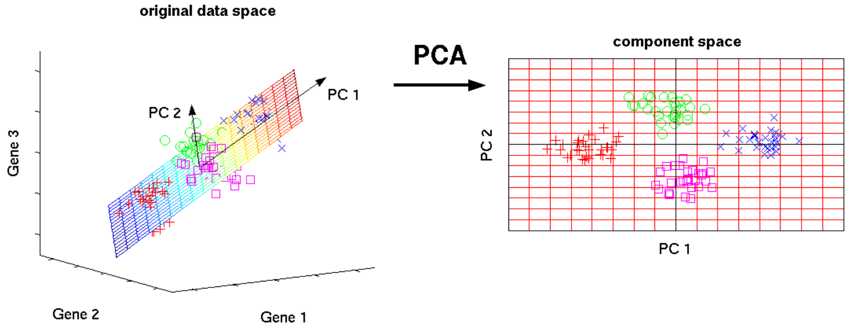


Figure 10. PCA Dimensional Reduction Visualization (Scholz, 2006)

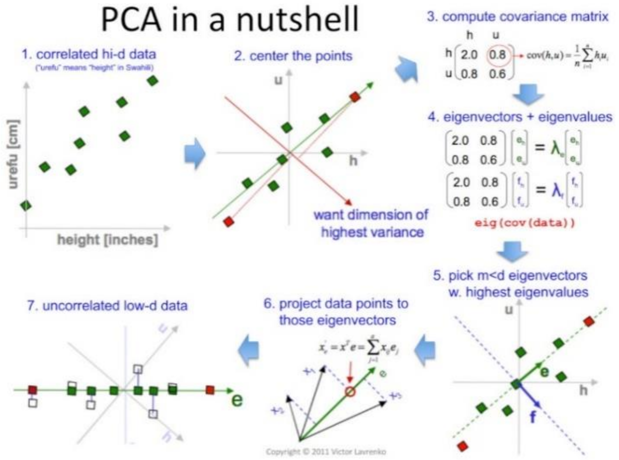

PCA in a nutshell
1. correlated hi-d data
urefu [cm]
height [inches]
2. center the points
u
h
want dimension of highest variance
3. compute covariance matrix
4. eigenvectors + eigenvalues
eig(cov(data))
5. pick m<d eigenvectors w. highest eigenvalues
6. project data points to those eigenvectors
7. uncorrelated low-d data
e
f
Copyright © 2011 Victor Lavrenko

Figure 11. PCA in nutshell (Laverenko and Sutton, 2011)

c. Artificial Neural Network (ANN) Model)

Artificial neural networks (ANN) using the **principles of biological neural network function** to estimate the correlation between a set of input and output parameters which can be used both in classification, categories, and regression problems, where the correlation between input and output is required (Karkalos and Markopoulos, 2017). This Unsupervised Method has one type called **Perceptron** with the outputs are 0 and 1. It is determined by the weighted summation $(w_j x_j)$ of the data inputs that have less or greater value than the threshold value.

In most cases of ANN, the basic method that we will use is the **The Multilayer Perceptron**, which has neurons (perceptrons) in the hidden layers for the deployment of weights and functions. The variation in weights and the threshold will get different models for decision-making. However, the ANNs generally suffer from **overfitting problems** because a reasonably sized ANN has far more parameters to be estimated than there are transfer samples available (Brown, 2009). That is why we should not use robust parameters, especially for the Model of Prediction.

$$output = \begin{cases} 0\ if\ \sum_j w_j x_j \leq threshold \\ 1\ if\ \sum_j w_j x_j > threshold \end{cases} \tag{8}$$

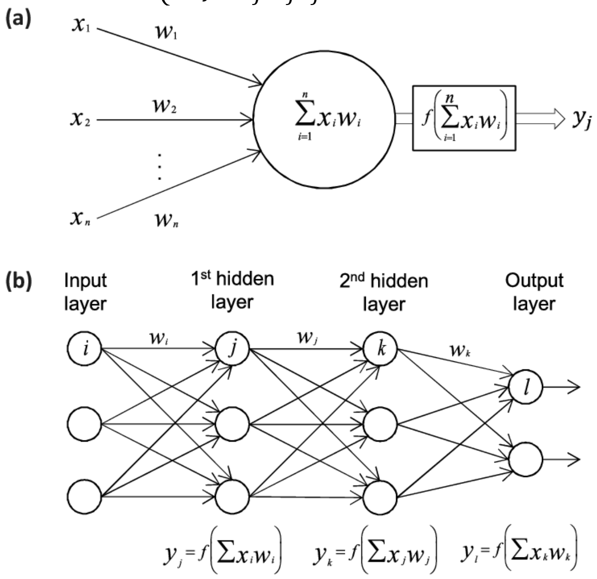


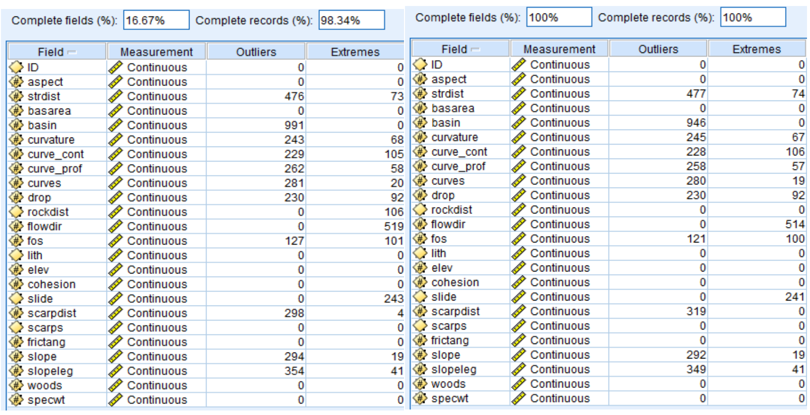
Figure 12. The visualization concept of Artifical Neural Network where the sum of x (input data) and w (weight) is passed through a particular non-linear activation function, to give an output y (Vieira et al., 2017)

4. DATA PROCESSING

4.1 Data Preparation

4.1.1 Data Cleaning

Given dataset was loaded into SPSS program and the initial step taken was set-up delimiter and storage type. Since the data consist of numerical values, both integer and real data type were chosen. The next step was removing all missing values by applying the data audit node to enhance the general data quality. The process of removal missing values were influenced the ROCKDIST parameter and made it only containing zero value, to prevent distortion in the modeling process, this parameter was filtered. One parameter needs to be corrected was SLIDE which contain values up to 10. By applying the "derive" tool, SLIDE data was transformed into a binary unit. Some parameters were removed to avoid redundancy in analysis. (see **Figure 14**).



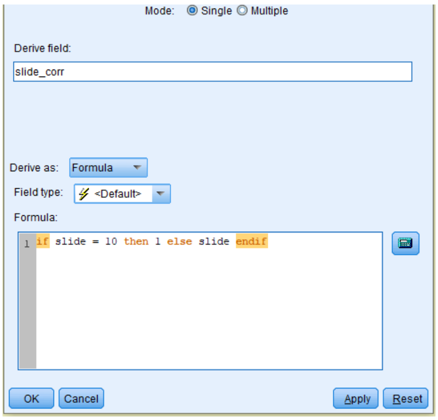
Complete fields (%): 16.67% Complete records (%): 98.34%

Field	Measurement	Outliers	Extremes
ID	Continuous	0	0
aspect	Continuous	0	0
strdist	Continuous	476	73
basarea	Continuous	0	0
basin	Continuous	991	0
curvature	Continuous	243	68
curve_cont	Continuous	229	105
curve_prof	Continuous	262	58
curves	Continuous	281	20
drop	Continuous	230	92
rockdist	Continuous	0	106
flowdir	Continuous	0	519
fos	Continuous	127	101
lith	Continuous	0	0
elev	Continuous	0	0
cohesion	Continuous	0	0
slide	Continuous	0	243
scarpdist	Continuous	298	4
scarps	Continuous	0	0
frictang	Continuous	0	0
slope	Continuous	294	19
slopeleg	Continuous	354	41
woods	Continuous	0	0
specwt	Continuous	0	0

Complete fields (%): 100% Complete records (%): 100%

Field	Measurement	Outliers	Extremes
ID	Continuous	0	0
aspect	Continuous	0	0
strdist	Continuous	477	74
basarea	Continuous	0	0
basin	Continuous	946	0
curvature	Continuous	245	67
curve_cont	Continuous	228	106
curve_prof	Continuous	258	57
curves	Continuous	280	19
drop	Continuous	230	92
rockdist	Continuous	0	0
flowdir	Continuous	0	514
fos	Continuous	121	100
lith	Continuous	0	0
elev	Continuous	0	0
cohesion	Continuous	0	0
slide	Continuous	0	241
scarpdist	Continuous	319	0
scarps	Continuous	0	0
frictang	Continuous	0	0
slope	Continuous	292	19
slopeleg	Continuous	349	41
woods	Continuous	0	0
specwt	Continuous	0	0

Figure 13. PCA in nutshell (Laverenko and Sutton, 2011)



Mode: ● Single ○ Multiple

Derive field: slide_corr

Derive as: Formula

Field type: <Default>

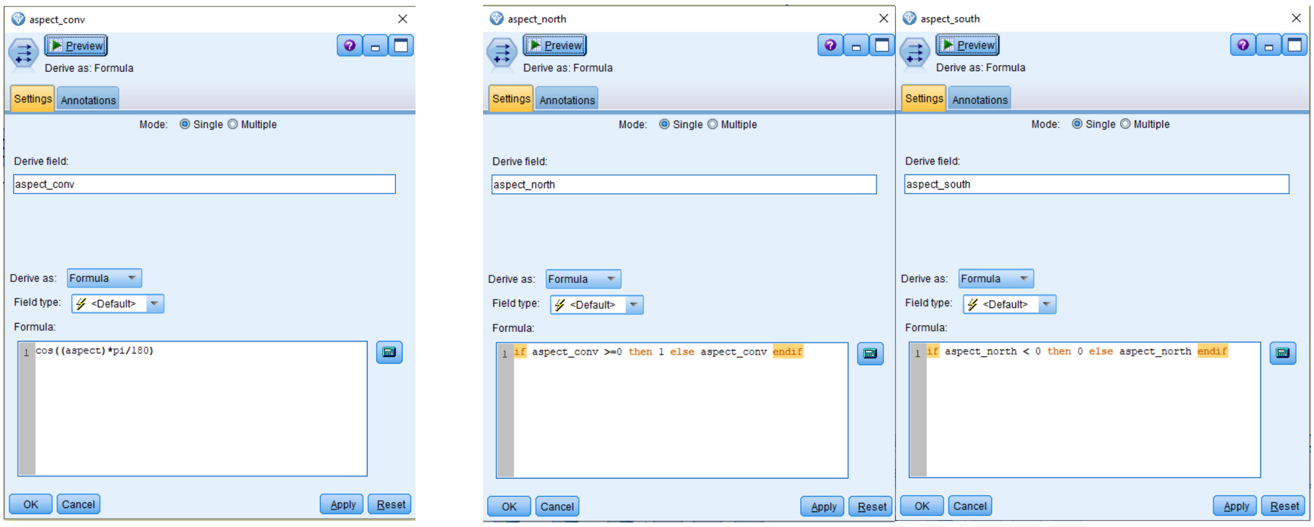
Formula:

```
if slide = 10 then 1 else slide endif
```

Figure 14. Slide Derive

4.1.2 Defining Aspect

The ASPECT parameter was converted from degrees into radians to correct value of slope exposition particularly in the flat area. The ASPECT then was divided into two separate areas, North and South. The north and south slope respectively consist of all positive and negative values. Therefore, ASPECT becomes a binary parameter, and the old values were filtered. (see **Figure 15**). The north and south slope respectively consist of all positive and negative values. Therefore, ASPECT becomes a binary parameter of **1 (North)** and **0 (South).** In Python-Based, we inspect first the relationship among our feature to get some insight for further analysis. Here we use the **Pearson Correlation** method which tell us about the linear relationship between two features, and have **value between -1 and 1**



aspect_conv

Derive as: Formula

Mode: ● Single ○ Multiple

Derive field: aspect_conv

Derive as: Formula

Field type: <Default>

Formula:

```
cos((aspect)*pi/180)
```

aspect_north

Derive as: Formula

Mode: ● Single ○ Multiple

Derive field: aspect_north

Derive as: Formula

Field type: <Default>

Formula:

```
if aspect_conv >=0 then 1 else aspect_conv endif
```

aspect_south

Derive as: Formula

Mode: ● Single ○ Multiple

Derive field: aspect_south

Derive as: Formula

Field type: <Default>

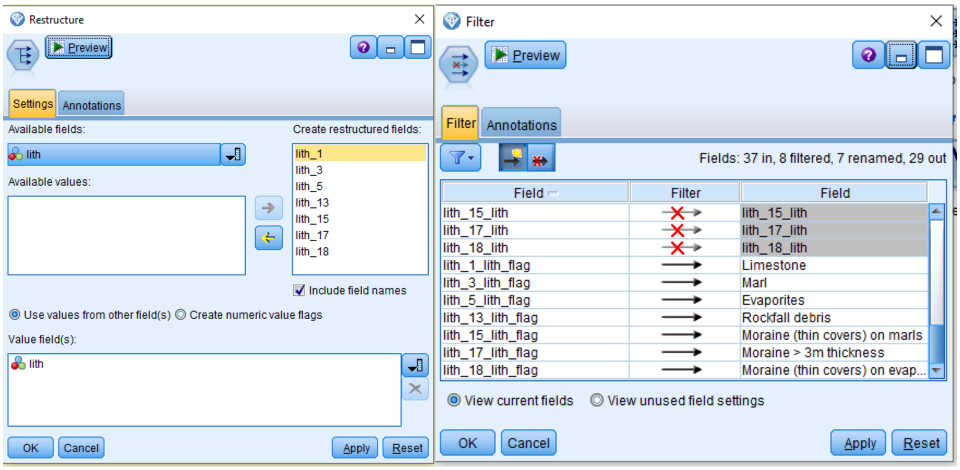
Formula:

```
if aspect_north < 0 then 0 else aspect_north endif
```

Figure 15. Defining Aspect North and South

4.1.3 Lithology Separation

The Lithology (LITH) separation was done by applying the node "restructure" for obtaining of new rock identities. Seven group of lithologies were created. To simplify modeling process, all lithologies data were converted to binary type and each "$null$" values were changed to 0. The old lithology parameters were removed, and the new binary parameters were renamed (see **Figure 16**).



Restructure

Available fields: lith

Available values:

Create restructured fields:
lith_1
lith_3
lith_5
lith_13
lith_15
lith_17
lith_18

☑ Include field names

● Use values from other field(s) ○ Create numeric value flags

Value field(s): lith

Filter

Fields: 37 in, 8 filtered, 7 renamed, 29 out

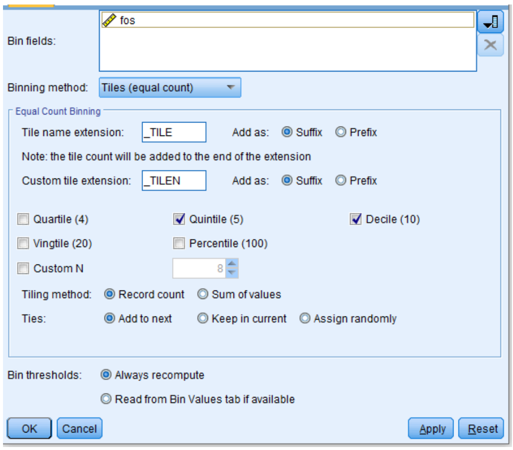
Field	Filter	Field
lith_15_lith	✗→	lith_15_lith
lith_17_lith	✗→	lith_17_lith
lith_18_lith	✗→	lith_18_lith
lith_1_lith_flag	→	Limestone
lith_3_lith_flag	→	Marl
lith_5_lith_flag	→	Evaporites
lith_13_lith_flag	→	Rockfall debris
lith_15_lith_flag	→	Moraine (thin covers) on marls
lith_17_lith_flag	→	Moraine > 3m thickness
lith_18_lith_flag	→	Moraine (thin covers) on evap...

● View current fields ○ View unused field settings

Figure 16. Lithology Separation

4.1.4 Factor of Safety (FoS) Classification

The Factor of Safety (FoS) of a slope is the ratio of resisting force to driving force. To capture the influence of FoS in more detail towards the model, the parameter was separated into 5 (quintile) and 10 (decile) data ranges (see **Figure 17**).



Bin fields: fos

Binning method: Tiles (equal count)

Equal Count Binning

Tile name extension: _TILE Add as: ● Suffix ○ Prefix

Note: the tile count will be added to the end of the extension

Custom tile extension: _TILEN Add as: ● Suffix ○ Prefix

□ Quartile (4) ☑ Quintile (5) ☑ Decile (10)

□ Vingtile (20) □ Percentile (100)

□ Custom N 8

Tiling method: ● Record count ○ Sum of values

Ties: ● Add to next ○ Keep in current ○ Assign randomly

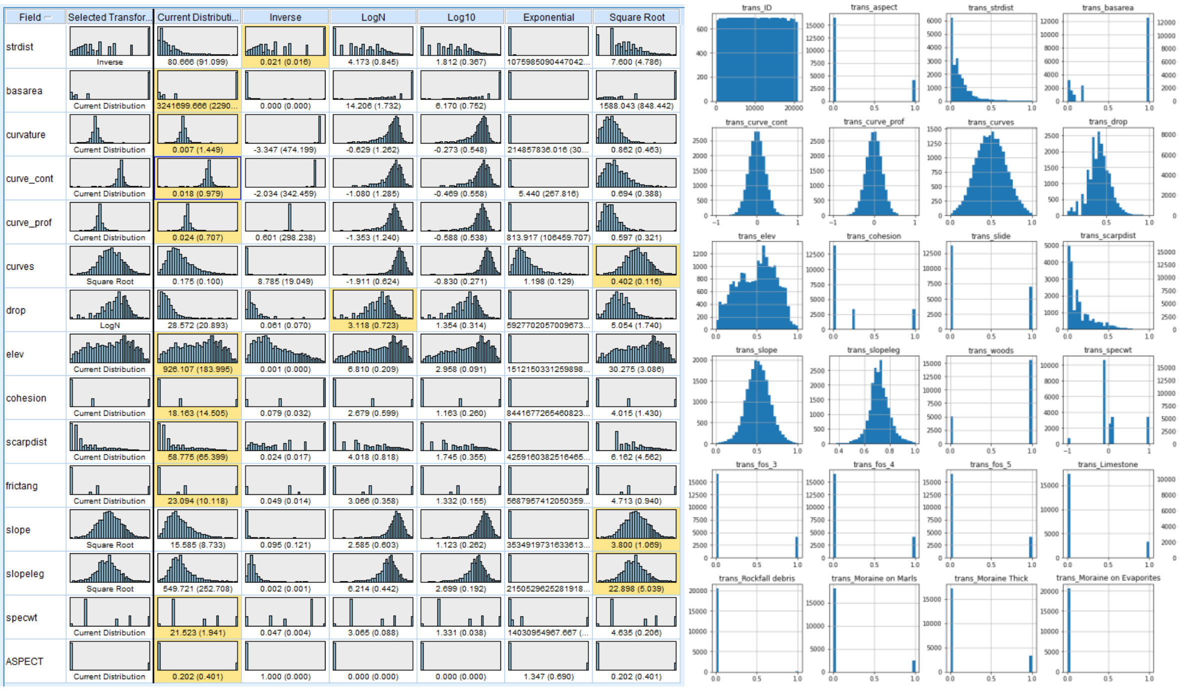
Bin thresholds: ● Always recompute

○ Read from Bin Values tab if available

Figure 17. FoS Binning

4.1.5 Transforming, Normalizing, and Re-Scaling

In this step, we do the data transformation into normal distribution data and data re-scaling to ease optimization process since normal distribution would reduce the heteroscedasticity. The data transformation could help a better result (model of prediction) when it is numerically in a normal distribution. The method to transform the data distribution, based on the Module in Scikit in Python, are Quantile Transform, Log, Yeo-Jhonson, and Box-Cox, in which we can decide it subjectively. In IBM SPSS, we have Inverse, LogN, Exponential, and Square Root. For the data re-scaling, we apply for data in each feature a range of either **(0 - 1)** or **((-1) - 1)**.

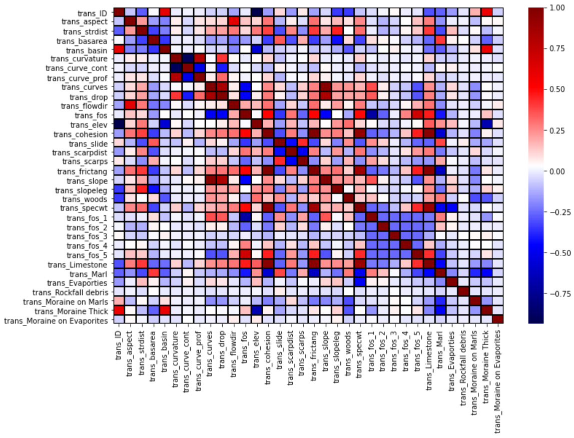


Field	Selected Transfor...	Current Distributi...	Inverse	LogN	Log10	Exponential	Square Root
strdist	Inverse	80.666 (91.099)	0.021 (0.016)	4.173 (0.845)	1.812 (0.367)	1075985090447042...	7.600 (4.786)
basarea	Current Distribution	3241699.666 (2290...	0.000 (0.000)	14.206 (1.732)	6.170 (0.752)		1588.043 (848.442)
curvature	Current Distribution	0.007 (1.449)	-3.347 (474.199)	-0.629 (1.262)	-0.273 (0.548)	214857836.016 (30...	0.862 (0.463)
curve_cont	Current Distribution	0.018 (0.979)	-2.034 (342.459)	-1.080 (1.285)	-0.469 (0.558)	5.440 (267.816)	0.694 (0.388)
curve_prof	Current Distribution	0.024 (0.707)	0.601 (298.238)	-1.353 (1.240)	-0.588 (0.538)	813.917 (106459.707)	0.597 (0.321)
curves	Square Root	0.175 (0.100)	8.785 (19.049)	-1.911 (0.624)	-0.830 (0.271)	1.198 (0.129)	0.402 (0.116)
drop	LogN	28.572 (20.893)	0.061 (0.070)	3.118 (0.723)	1.354 (0.314)	5927702057009673...	5.054 (1.740)
elev	Current Distribution	926.107 (183.995)	0.001 (0.000)	6.810 (0.209)	2.958 (0.091)	1512150331259898...	30.275 (3.086)
cohesion	Current Distribution	18.163 (14.505)	0.079 (0.032)	2.679 (0.599)	1.163 (0.260)	8441677265460823...	4.015 (1.430)
scarpdist	Current Distribution	58.775 (65.399)	0.024 (0.017)	4.018 (0.818)	1.745 (0.355)	4259160382516465...	6.162 (4.562)
frictang	Current Distribution	23.094 (10.118)	0.049 (0.014)	3.066 (0.358)	1.332 (0.155)	5687957412050359...	4.713 (0.940)
slope	Square Root	15.585 (8.733)	0.095 (0.121)	2.585 (0.603)	1.123 (0.262)	3534919731633613...	3.800 (1.069)
slopeleg	Square Root	549.721 (252.708)	0.002 (0.001)	6.214 (0.442)	2.699 (0.192)	2150529625281918...	22.898 (5.039)
specwt	Current Distribution	21.523 (1.941)	0.047 (0.004)	3.065 (0.088)	1.331 (0.038)	14030954967.667 (...	4.635 (0.206)
ASPECT	Current Distribution	0.202 (0.401)	1.000 (0.000)	0.000 (0.000)	0.000 (0.000)	1.347 (0.690)	0.202 (0.401)

Figure 18. Transformation, Normalization, and Scaling in IBM SPSS (**Left**) and Python (**Right**)

4.1.5 Pearson Correlation for Choosing Features for Input Data of Modelling

In Python-based, we want to get an insight about the distribution of Pearson's Correlation value among all features in input data, in order to get a statistically representative features for Landslide Prediction. Then, we set a threshold value of the Pearson Correlation between all features with the Landslide data so that can be assigned for further Modelling. We also take into account the P-Value for all the correlation values. We set threshold value of **Pc > 0.3** and **Pc < -0.3** and we get:

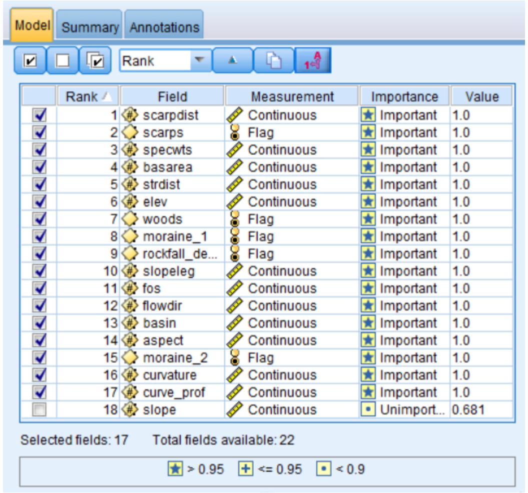


No.	Variable with trans_slide	Pearson Value
1	trans_slide	1.000.000
2	trans_scarps	0.438180
3	trans_Marl	0.381935
4	trans_basarea	0.345430
5	trans_cohesion	-0.306148
6	trans_strdist	-0.307812
7	trans_frictang	-0.312783
8	trans_fos	-0.353007
9	trans_scarpdist	-0.436005

Figure 19. Pearson Correlation of Data Input (**Left**) and all the features for Input Data of Modelling in Python-Based.

4.1.6 Data Partitioning (Training Data and Test Data).

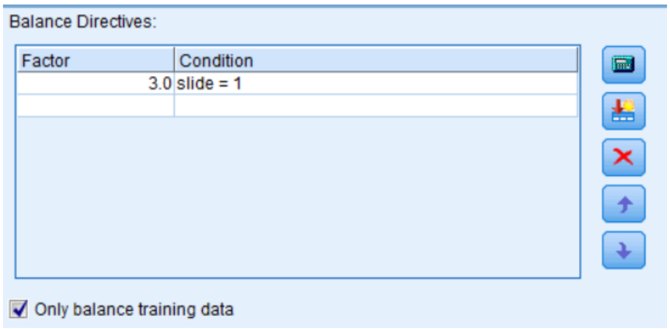
In order to create the model of machine learnings later phyton-based project, we partition our data into Training Data and Test Data. In this phyton-based project, we partition 60% for training data and 40% for test data. In IBM SPSS-based project, we partition 70% for training data and 30% for test data, which is also sorted according to importance to the model later (in this case, we assign 17 features) and being specified for importance value only > 0.95. Moreover, The Balancing in IBM SPSS is also mandatory to improve the accuracy of the training model by multiplied 3 for a landslide to artificially raised the appearance in the modelling.



☐	Rank	Field	Measurement	Importance	Value
☑	1	scarpdist	Continuous	Important	1.0
☑	2	scarps	Flag	Important	1.0
☑	3	specwts	Continuous	Important	1.0
☑	4	basarea	Continuous	Important	1.0
☑	5	strdist	Continuous	Important	1.0
☑	6	elev	Continuous	Important	1.0
☑	7	woods	Flag	Important	1.0
☑	8	moraine_1	Flag	Important	1.0
☑	9	rockfall_de...	Flag	Important	1.0
☑	10	slopeleg	Continuous	Important	1.0
☑	11	fos	Continuous	Important	1.0
☑	12	flowdir	Continuous	Important	1.0
☑	13	basin	Continuous	Important	1.0
☑	14	aspect	Continuous	Important	1.0
☑	15	moraine_2	Flag	Important	1.0
☑	16	curvature	Continuous	Important	1.0
☑	17	curve_prof	Continuous	Important	1.0
☐	18	slope	Continuous	Unimport...	0.681

Selected fields: 17 Total fields available: 22

★ > 0.95 ⊞ <= 0.95 ▣ < 0.9


Balance Directives:

Factor	Condition
3.0	slide = 1

☑ Only balance training data

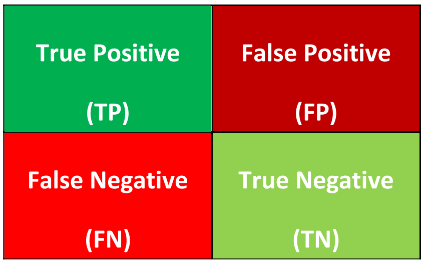
Figure 20. Pearson Correlation of Data Input (**Left**) and all the features for Input Data of Modelling in Python-Based

After we are ready and clear with the Data Preparation for this phyton-based project, we can develop several models for Landslide Prediction using several chosen features. As a basic Machine Learning concept, we know that the models are just estimator tools based on observed data. However, such models could provide us a mathematically objective construction of how the later new input data of features will indicate a Landslide or not.

4.2 Machine Learning Model Creation and Evaluation

After we are cleared with the Data Preparation for this IBM SPSS- based and Phyton-based machine learning project in Landslide Susceptibility, we develop several models for it like in **Figure 4.** One thing that should be noted, as a basic Machine Learning concept, we know that the models are just estimator tools based on the input data (observed data) only, which might or might not be representative for the further new data input. However, such models could provide us a mathematically objective construction of how the recent input data have created a Machine Learning system for later new input data of features to indicate or estimate the Landslide Susceptibility of particular conditions.

Therefore, we applied several models for both the Supervised Model and the Unsupervised Model for Landslide Prediction, with a variation or model scheme of Multivariate and Bivariate case, in IBM SPSS and Python project. Afterward, to confirm that our models are representative enough to be used later for Landslide Prediction, we also do the machine learning models evaluations of Confusion Matrix, Score Accuracy, Precision, Recall, F1 Score, and Receiver Operating Characteristic (ROC). **Precision** explains how precise/accurate our model is from those predictive positives, how many of them are actual positive. Meanwhile, **Recall** informs about how many of the actual positives were captured by our model which was identified as positive. **F1 Score** is needed when a balance between Precision and Recall was pursued. **Accuracy** can tell us immediately whether a model is being trained correctly and how it may perform generally. However, it does not always give detailed information regarding its application to the problem.



True Positive (TP)	False Positive (FP)
False Negative (FN)	True Negative (TN)

$$Score\ Accuracy = \frac{TN + TP}{TN + FP + TP + FN}$$

$$Precision = \frac{TP}{FP+TP} \qquad Recall = \frac{TP}{TP+FN}$$

$$F1\ Score = 2 * \frac{Precision * Recall}{Precision + Recall}$$

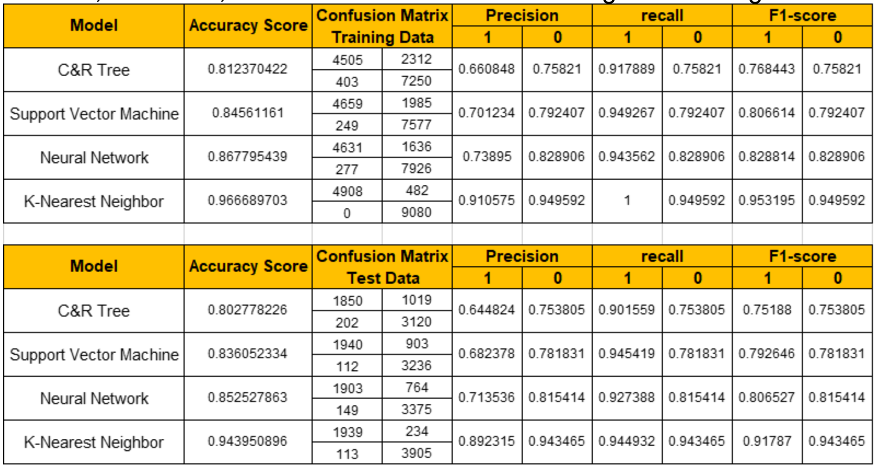
Figure 20. Confusion Matrix and the calculation for Score Accuracy, Precision, Recall, and F1 Score.

5. RESULTS AND DISCUSSION

5.1 Confusion Matrix

Four different models consist of Classification and Regression (C&R) Tree, SVM, Neural Network and K-Nearest Neighbor (KNN) were run through the dataset using IBM SPSS. The modeling quality was tested using the Confusion Matrix, Score Accuracy, Precision, Recall, and F1 score (**Table 2**). As shown in the table, case 1 represents when a landslide is predicted to occur, while case 0 describes the opposite. For example, in the model of C&R Tree for testing data, the model respectively gave 0.66 and 0.76 precision accuracy for case 1 and case 0. It can be interpreted as a 66% chance that a landslide will occur when we initially predicted that a landslide would happen. Otherwise, when we initially predicted that a landslide would not happen, there is a 76 % chance that a landslide would not happen in an actual event landslide. In the case of recall interpretation for the C&R Tree model through training data, 92 % can be interpreted as in case of landslide actually occurred, how accurate that we predicted it before. As the opposite case, when landslide actually not happened, how accurate also that we predicted initially. The way we interpret such evaluation indicators could be applied to all model that being listed in **Table 2**. Moreover, the interpretation can be applied for other models both in training and testing data sets. Basically, for a good classifier, each parameter; Precision, Recall, F1 Score, and Score Accuracy, should ideally be as high as 1. Ultimately, the K-Nearest Neighbor has the highest value in all categories, both in training and test data compare to other models in IBM SPSS.

Table 2. Confusion Matrix, Precision, Recall and F1 Score for Training and Testing Data in IBM SPSS Project.



Model	Accuracy Score	Confusion Matrix Training Data		Precision		recall		F1-score	
				1	0	1	0	1	0
C&R Tree	0.812370422	4505	2312	0.660848	0.75821	0.917889	0.75821	0.768443	0.75821
		403	7250						
Support Vector Machine	0.84561161	4659	1985	0.701234	0.792407	0.949267	0.792407	0.806614	0.792407
		249	7577						
Neural Network	0.867795439	4631	1636	0.73895	0.828906	0.943562	0.828906	0.828814	0.828906
		277	7926						
K-Nearest Neighbor	0.966689703	4908	482	0.910575	0.949592	1	0.949592	0.953195	0.949592
		0	9080						

Model	Accuracy Score	Confusion Matrix Test Data		Precision		recall		F1-score	
				1	0	1	0	1	0
C&R Tree	0.802778226	1850	1019	0.644824	0.753805	0.901559	0.753805	0.75188	0.753805
		202	3120						
Support Vector Machine	0.836052334	1940	903	0.682378	0.781831	0.945419	0.781831	0.792646	0.781831
		112	3236						
Neural Network	0.852527863	1903	764	0.713536	0.815414	0.927388	0.815414	0.806527	0.815414
		149	3375						
K-Nearest Neighbor	0.943950896	1939	234	0.892315	0.943465	0.944932	0.943465	0.91787	0.943465
		113	3905						

Next, in Python-based, we conduct 12 models with the same evaluation test indicator as in IBM SPSS and summarized in **Table 3.** In this case, the Multi-Layered Perceptron (MLP model) brings us the most significant evaluation among other methods. However, as the MLP is very prone to overfitting when new data is acquired, we think Multivariate SVM with kernel RBF should be the model that we can take a reasonable consideration. Decision Tree is also a promising model that give us a good result, in the basis of entropy calculation, for Landslide Susceptibility. Moreover, one interesting thing here is how PCA's dimensional reduction technique could successfully enhance the Accuracy score for K-Means from 0.3 to 0.6. Even the PCA could not improve MLP, at least the use of PCA could be a solution for enhancing the model for other Machine Learning methods. Logistic Regression also has a good Accuracy Score that can be used as an alternative for the categorical Machine Learning model, even with the basis of regression line in Landslide Susceptibility. The general assessment for Multivariate and Bivariate models Accuracy score, lead us to the conclusion that the enhancement of Machine Learning model could parallel with more variables we took into considerations, except for K-Means. In the end, we think that all models that give us an Accuracy Score of more than 0.8, either in IBM SPSS or Python, could be evidently used for Landslide Susceptibility Machine Learning

Table 3. Confusion Matrix, Precision, Recall and F1 Score for Python-Based Project.

Model	Accuracy Score	Confusion Matrix		Precision		recall		F1-score	
				1	0	1	0	1	0
Multi Layered Perceptron (MLP)	0.838113	4809	669	0,87787514	0,75995694	0,87787514	0,75995694	0,87787514	0,75995694
		669	2118						
Multivariate SVM RBF	0.837629	4751	727	0,86728733	0,74922387	0,88538949	0,74922387	0,87624493	0,74922387
		615	2172						
Decision Tree	0.830732	4718	760	0,86126323	0,73865199	0,88071682	0,73865199	0,8708814	0,73865199
		639	2148						
Multivariate SVM Linear	0.817907	4771	707	0,8709383	0,73775964	0,85809353	0,73775964	0,8644682	0,73775964
		789	1989						
Logistic Regression	0.815971	4789	689	0,87422417	0,73940998	0,85198363	0,73940998	0,86296063	0,73940998
		832	1955						
MLP with PCA	0.788627	4886	592	0,89193136	0,73381295	0,80880649	0,73381295	0,84833753	0,73381295
		1155	1632						
Bivariate SVM Linear	0.768905	4484	994	0,81854691	0,6530541	0,83037037	0,6530541	0,82441625	0,6530541
		916	1871						
Bivariate SVM RBF	0.768784	4762	716	0,86929536	0,6897747	0,79939567	0,6897747	0,8328815	0,6897747
		1195	1592						
Bivariate K-Means	0.685736	8244	5457	0,6017079	0,52051665	0,88836207	0,52051665	0,71746225	0,52051665
		1036	5924						
Multivariate K-Means with PCA	0.639971	1400	1324	0,51395007	0,48462437	0,89514066	0,48462437	0,65298507	0,48462437
		164	1245						
Multivariate K-Means	0.322928	2078	3400	0,37933552	0,14808319	0,4861956	0,14808319	0,42616899	0,14808319
		2196	591						
Bivariate SVM Polynomial	NaN	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-

5.2 Profit Diagram

Another method to evaluate the model quality is by using profit models which can be generated with IBM SPSS Modeler. In this method, the testing data set is compared with the training data set by plotting the profit of both next to each other. The better congruency of both graphs, the better is the models learning effect. **Figure 21** shows the profit diagrams for all used models.

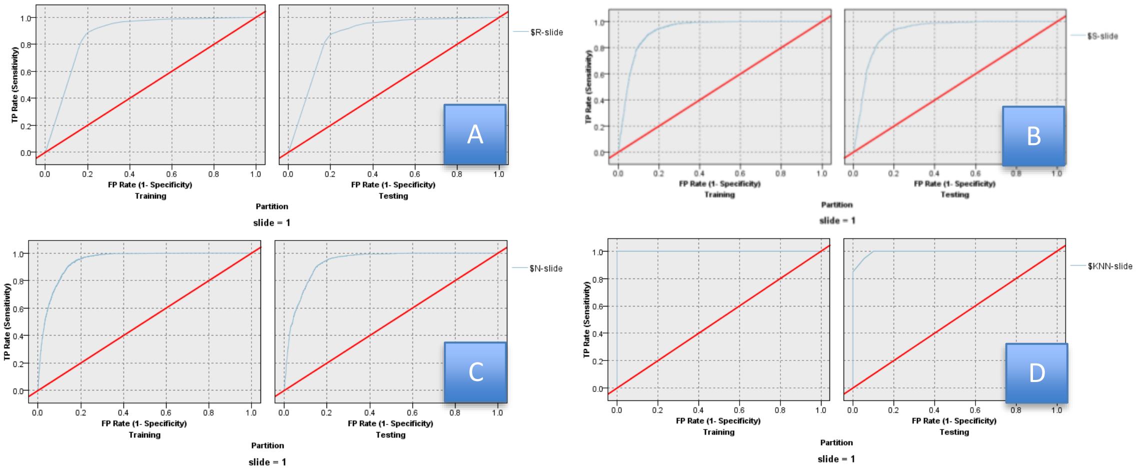


Figure 21. Profit Diagram for C&R Tree (**A**), SVM (**B**), Neural Network (**C**), and KNN (**D**)

5.3 Importance Features for Machine Learning Model

Regarding the features that contribute to the construction of the Machine Learning Model, IBM SPSS come up with the result that can vary from application of one model to another. Here in **Figure 22.** We could see that for SVM, and C&R Tree, scarpdist, specwts, basarea, and Moraine_1 give the most significant contributor. However, for the Neural Network, fos, scarpdist, and curve_prof are the top three contributors for the model. On the other hand, the significant contributors in Python-based could be assessed before modeling the machine learning from the Pearson Coefficient value. Based on **Figure 19,** we acquire the information that scarpdist, scarps, specwts, Marl, and fos took the high contributions toward the Machine Learning Modeling in Python Based.

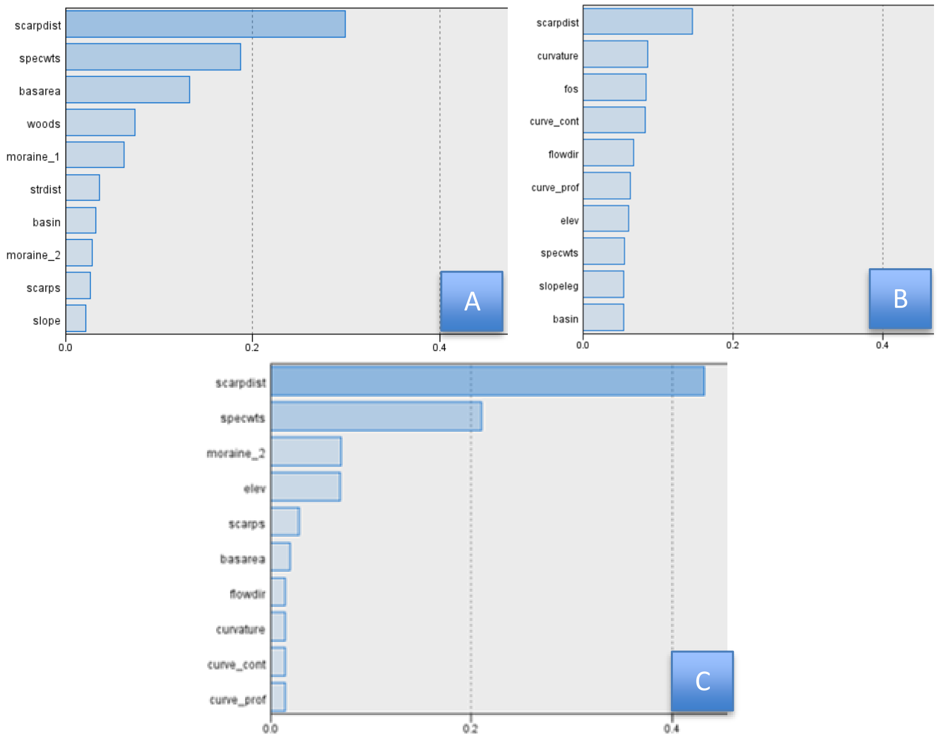


Figure 22. Predictor Importance on the basis of **(A)** SVM, **(B)** Neural Network, and **(C)** C&R Tree.

Hence, from both IBM SPSS and Python, we see that the value of distance to failure scarps (scarpdist), Specific weight (specwts), with lithology of Moraine or Marl have the major role to the Machine Learning Model. In a simple explanation, Scarp is a very steep (almost vertical) slope region in which soil or rock is exposed due to the failure surface, faulting, or erosion (**Figure 23**) associated with Landslide. Therefore, based on the Pearson Value in **Figure 19**, the feature scarp is highly positive (Pc = 0.4), which means positively correlated. Moreover, the value for scarpdist is highly negative (Pc = - 0.4), which means inversely related to the Landslide event. It is, of course, geologically make sense that the closer location of the data to the area of the Scarp feature, the more such location to slide as the gravity force in rock material. Publication from Gorum et al. in 2011 also stated that Landslide density will increases in particular areas where relatively close to the fault rupture, and with high relief and slope gradient which associated with scarps.

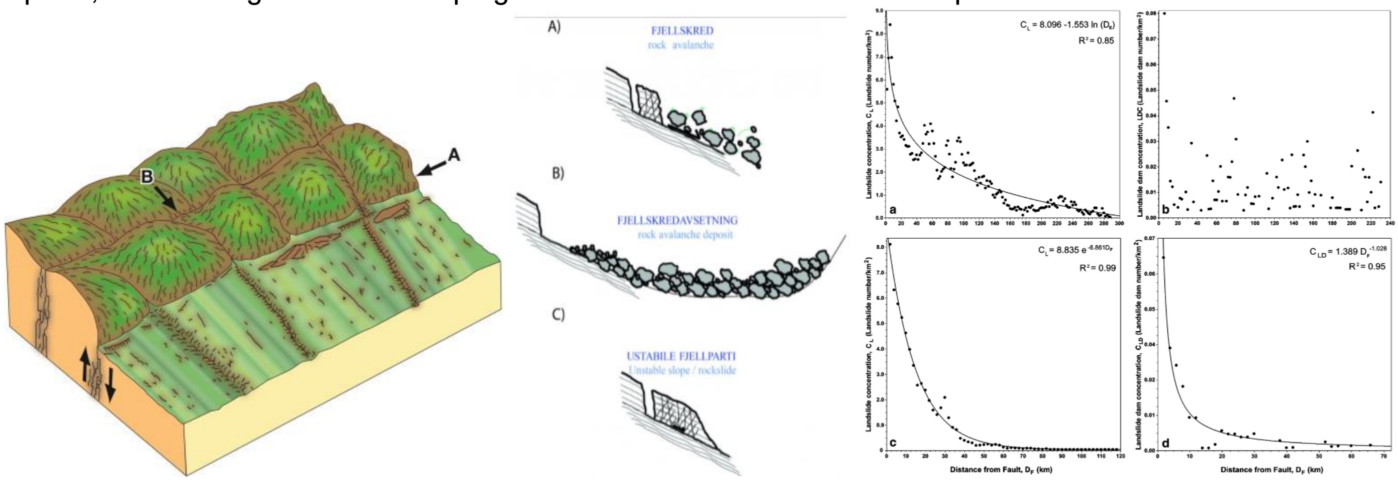


Figure 23. (LEFT) A fault scarp (**A**) results directly from fault displacement whereas a fault-line scarp results from differential erosion (**B**) (Nash, 2013). (**CENTER**) Rock Avalanche event (ngu.no, 2020). (**RIGHT**) Relationship between landslide (dam) concentration and distance from epicenter and fault. a: Distance from epicenter for all landslides; b: distance from epicenter forlandslide dams; c: distance from fault rupture for all landslides; and d: distance from fault rupture for landslide dams (Gorum et al., 2011).

By definition, specific weight means the weight (force due to gravitational) per unit volume (N/m^3), depending on the gravitational field. Hence, the more a particular rock constituent have their specific weight, landslide event should logically tend to happen as the weight force due to gravity is increased in that area, in which also proven from the Predictor Importance graph in **Figure 22.** The high rock avalanche (**Figure 23**) during the 2008 hazard event on our study site could also be the prominent evidence of how specific weight took a significant role in landslides.

Moraine can be explained as an accumulation of unconsolidated rock material or debris (regolith or soil and rock) formed by the carrying of glacier ice. In the sense of study site, it is predicted that such rock formed from Tibetan Plateau. Based on the publication by Gan and Zhang in 2019, it is stated that the landslide rock debris is commonly related to the broken rock from bedrock which might be associated with the Moraine. The Predictor Importance in **Figure 22** should be good evidence for such rock contributed to the Landslide Susceptibility.

The highly positive Pearson value between Marlstone and landslide from **Figure 19** also favors the feature that has a significant role on Landslide Susceptibility. The natural landslide on the marl-rich, by default, presents limited stability, in which the stability will decrease more during heavy rainfall (Christaras et al., 2014). We have discussed that the climate in The Longmen mount is a humid subtropical climate, and heavy rainfall is a common thing during a particular time (Gan and Zhang, 2019). Therefore, Marlstone should be the feature that is associated positively with the Landslide Susceptibility.

Figure 24. (LEFT) Marlstone and **(RIGHT)** (Christaras et al., 2014).
(RIGHT) Moraines Rock (Landcareresearch.co.nz, 2021)

5.4 Specific Task

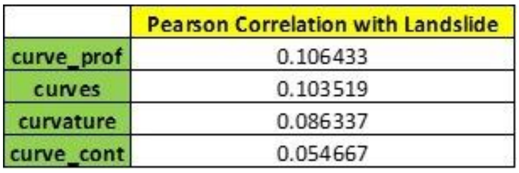
Curves, Curvature, Curvature Contour and Curvature Profile.

Curves can be defined as the change in height derived from the slope, while curvature measures how curved a curve is or a distance from being a straight line (Abate and Tovena 2012, as cited in Krebs et al., 2015). To measures the curvature, several approaches are used including curvature contour and curvature profile, which respectively measure horizontal and vertical planes of slope curvature.

To measure the influence of several curvature variables towards landslide susceptibility in our study, the Pearson Value for all curvature variables (features) was extracted as seen in **Table 4**. Based on the provided table, statistically the correlation values of curvature variables towards the landslide are not high with value between 0.05 – 0.1. It also can be interpreted that among curvature variables, curve profile and curves are the most significant factor for landslide event occurrence.

As a result of the modeling process with IBM SPSS, particularly with C&R Tree and Neural Network modeling, curvature variables appear on the list of Predictor Importance (**see Figure 22**). Consistent with the Pearson Correlation result, the value of curvature, curvature contours and profiles are below 0.2. In summary, it can be concluded that curvature variables delivered influences towards landslide susceptibility in our data.

Table 4. Pearson Correlation with Landslide



	Pearson Correlation with Landslide
curve_prof	0.106433
curves	0.103519
curvature	0.086337
curve_cont	0.054667

Zero points problem arising from the cosinus-transformation. How did you solve the problem. The common method to solve the problem is by assigning the zero point into north or south sectors. In our case, zero points were assigned to the north sector.

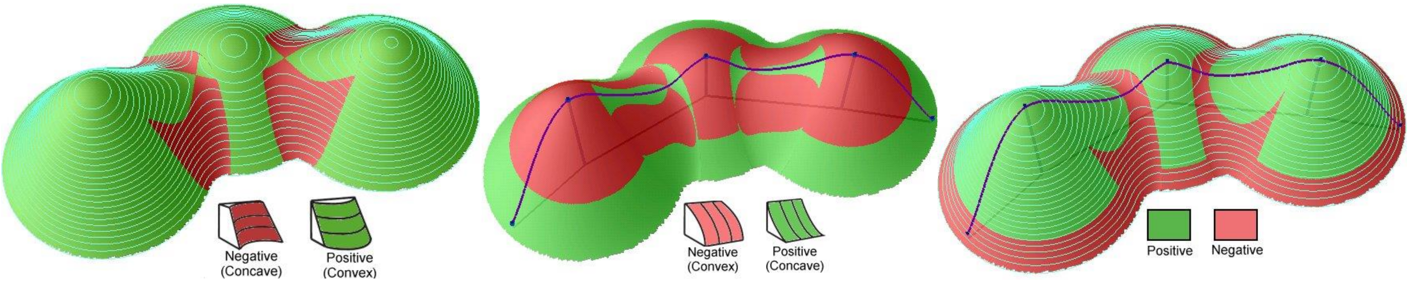


Figure 25. (**LEFT**) Plan or contour curvature, (**CENTER**) Profile Curvature, (**RIGHT**) Combined Curvature (ian-ko.com, 2021)

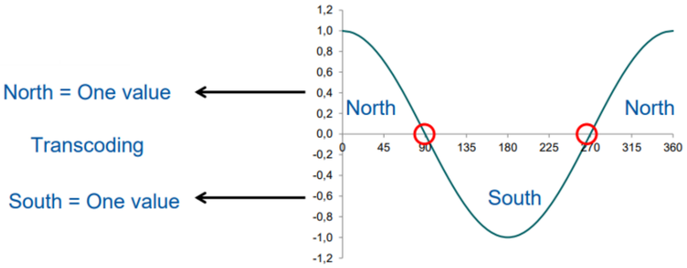


Figure 26. Cosinus Transformation for N/S Aspect (Dufresne et al., 2019)

REFERENCES

Anonymous, 2021. Teach you how to understand decision trees: from concept to application. https://www.programmersought.com/article/19876041005/. retrieved 28 July 2021.

Anonymous, 2021. Rock Avalanches in Norway. https://www.ngu.no/en/topic/rock-avalanches-norway. Retrieved 29 July 2021.

Anonymous, 2021. Moraines. https://www.landcareresearch.co.nz/publications/naturally-uncommon-ecosystems/inland-and-alpine/moraines/. Retrieved 29 July 2021.

Arentze, T.A., 2009. Spatial Data Mining, Cluster and Pattern Recognition. *International Encyclopaedia of Human Geography*. Elsevier. Pp 325 – 441. https://doi.org/10.1016/B978-008044910-4.00524-1.

Christaras, B., Argyriadis, M. & Moraiti, E., 2014. Landslides in the marly slope of the Kapsali area in Kithira Island, Greece. *Bull Eng Geol Environ* **73,** 839–844. https://doi.org/10.1007/s10064-013-0502-7

Cui, P., Chen, XQ., Zhu, YY. *et al.* 2011.The Wenchuan Earthquake (May 12, 2008), Sichuan Province, China, and resulting geohazards. *Nat Hazards* **56,** 19–36. https://doi.org/10.1007/s11069-009-9392-1

Brown, S., D., 2009. Comprehensive Chemometrics (Transfer of Multivariate Calibration Models). Elsevier. Pp 345 - 378.

Bzdok, D., Krzywinski, M., and Altman, N., 2018. Machine learning: Supervised methods. Nature Methods. 15. 10.1038/nmeth.4551.

Chen, L., 2019. *Support Vector Machine - Simply Explained*. https://towardsdatascience.com/support-vector-machine-simply-explained-fee28eba5496. Retrieved 29 July 2021.

Dufresne, A., Wellmann, F., Mouleifard, S., and Chudalla, 2021. Machine Learning in the Geosciences Lecture: Data Preparation. RWTH Aachen.

Edgar, T.W., and Manz, D.O., 2017. Research Methods for Cyber Security: Machine Learning. Syngress. Pp 153-173. https://doi.org/10.1016/B978-0-12-805349-2.00006-6.

Wellmann, F. and Chudalla, N., 2021. Exercise Python Module: 04_KMeans. RWTH Aachen.

Farber, R., 2012. CUDA Application Design and Development. Morgan Kaufmann. https://doi.org/10.1016/C2010-0-69090-0.

Gan, J., and Zhang, Y.X., 2019. Failure modes of loose landslide deposits in 2008 Wnchuan earthquake area in China. Nat. Hazards Earth Syst. Sci. Discuss. https://doi.org/10.5194/nhess-2019-25

Gorum, T., Fan, X., van Westen, C.J., Huang, R.Q., Xu, Q., Tang, C., Wang, G., 2011. Distribution pattern of earthquake-induced landslides triggered by the 12 May 2008 Wenchuan earthquake. Geomorphology. Volume 133, Issues 3-4. Pp 152 – 167. https://doi.org/10.1016/j.geomorph.2010.12.030.

Tchoukanski, I., 2021. Raster Curvature. https://www.ianko.com/ET_Surface/userguide/. Retrieved 29 July 2021

Jauregui, A.F., 2021. *How to code a logistic regression in R from scratch.* https://anderfernandez.com/en/blog/code-logistic-regression-r-from-scratch/. Retrieved 28 July 2021.

Karkalos, N. E. and Markopoulos, A. P., 2017. Modelling of hard machining. Woodhead Publishing Reviews: Mechanical Engineering Series. https://doi.org/10.1016/B978-0-85709-481-0.00006-9.

Krebs, P., Stocker, M., Pezzatti, G. B., and Conedera, M., 2015. An alternative approach to transverse and profile terrain curvature. *International Journal of Geographical Information Science.* http://dx.doi.org/10.1080/13658816.2014.995102.

Lavrenko, V. and Sutton, C., 2011. IAML: Dimensionality Reduction. School of informatics, University of Edinburgh.

Napoli, A., 2005. Handbook of Categorization in Cognitive Science: A Smooth Introduction to Symbolic Methods for Knowledge Discovery. Elsevier Science. Pp 913-933. https://doi.org/10.1016/B978-008044612-7/50096-2.

Nash, D., 2013. Treatise on Geomorphology: Tectonic Geomorphology of Normal Faults Scarps. Academic Press. Pp 234-249. https://doi.org/10.1016/B978-0-12-374739-6.00090-7

Raschka, S., 2021. *How to Select Support Vector Machine Kernels. https://www.kdnuggets.com/2016/06/select-support-vector-machine-kernels.html.* Retrieved 27 July 2021.

Rotondo, A., and Quilligan, F., 2020. Evolution Paths for Knowledge Discovery and Data Mining Process Models. SN Comput. Sci. 1, 109. https://doi.org/10.1007/s42979-020-0117-6.

Subasi, A., 2020. Practical Machine Learning for Data Analysis Using Python. Academic Press. https://doi.org/10.1016/C2019-0-03019-1.

Traverso, A., Dankers, F. J. W. M., Osong, B., Wee, L., and van Kuijk, S., M., J., 2018.Fundamentals of Clinical Data Science (Chapter 9: Diving Deeper into Models). Cham (CH): Springer. doi: 10.1007/978-3-319-99713-1_9

Scholz, M., 2006. Approaches to analyse and interpret biological profile data. Universitätsbibliothek Potsdam. urn:nbn:de:kobv:517-opus-7839.

Vieira, S., Pinaya, W., H., L., and Mechelli, A., 2017. Using deep learning to investigate the neuroimaging correlates of psychiatric and neurological disorders: Methods and applications. Neuroscience & Biobehaviorial Reviews 74. 58-75.

APPENDIX: Features Transformation and Normalization

Parameters	Transformation in IBM SPSS	Transformation in Python	Normalization
ID	-	-	-
ASPECT	-	-	0-1
STRDIST	Inverse	-	(-1) - 1
BASAREA	-	-	0 - 1
BASIN	Catchment	-	-
CURVATURE	-	Quantile Transformation	(-1) - 1
CURVE_CONT	-	Quantile Transformation	(-1) - 1
CURVE_PROF	-	Quantile Transformation	(-1) - 1
CURVES	Square Root	Yeo-Jhon Transformation	0 - 1
DROP	LogN	Box-Cox Transformation	0 - 1
ROCKDIST	-	-	0 - 1
FLOWDIR	-	-	-
FOS	-	Quantile Transformation	(-1) - 1
LITH	-	-	-
ELEV	-	-	0 - 1
COHESION	-	-	0 - 1
SLIDE	-	-	-
SCARPDIST	-	-	0 - 1
SCARPS	-	-	-
FRICTANG	-	-	0 - 1
SLOPE	-	Yeo-Jhon Transformation	0 - 1
SLOPELEG	Square Root	Box-Cox Transformation	0 - 1
WOODS	-	-	-
SPECWT	-	Log Transformation	0 - 1