Dear Anonymous Referee #1,

Thank you for your positive and constructive comments. Below is a documented list of changes we have made to the manuscript (marked R: in blue font). We have shortened the introduction of the ML algorithms, detailed the majority and minority class sampling used in the training and testing, revised the results by 5-fold cross-validation, and refined the discussion and conclusion of the findings. We hope these clarifications will improve the reader's understanding of our work.

Kind Regards,
Praveen Kumar

Anonymous Referee #1 Comments

1. The different ML algorithms are individually reported and described in perhaps too much detail.

   R: We appreciate the reviewer's feedback on our ML algorithm descriptions. In response, we have revised and shortened the introductions of each algorithm. We now provide a brief overview of each algorithm in three to four sentences, followed by an explanation focused solely on the crucial parameter variations and their significance in our experimental setup.

2. The monitored landslide is presented only in geographical terms. It might be useful to provide more details on the characteristics of the landslide.

   R: Thank you for this suggestion. We have revised the main manuscript and added the paragraph in the Data Collection and Description section on page 4 as follows, along with a new reference:

   "The monitored landslides are characterized as shallow landslides with debris flow, occurring at elevations ranging from 1450 m to 1920 m. The slopes in the landslide zones in the upper parts are made up of weathered limestone and dolomitic limestone, whereas the lower slopes exhibit black carbonaceous slate. The slates are highly weathered and leached, adorned with white and yellow encrustation. These are covered with a thin veneer of debris, mainly consisting of pebble- and cobble-sized limestone, sandstone, and slate embedded in a sand–silt–clay matrix. Additional context includes an annual rainfall of 4190 mm in the area, as reported by Gupta et al. (2015)."

   We added a new reference.

   Gupta, V., Bhasin, R. K., Kaynia, A. M., Tandon, R. S., & Venkateshwarlu, B. (2016). Landslide hazard in the Nainital township, Kumaun Himalaya, India: the case of September 2014 Balia Nala landslide. *Natural Hazards, 80*, 863-877.

3. However, it is not clear if the other two minority classes were oversampled, or if they were removed in subsequent analyses.

R: Thank you for your comment. We appreciate your feedback and have addressed this concern in the revised manuscript.

In the Class Labeling section on page 5, we now provide detailed information on the distribution of classes, explicitly stating the percentages for each category:

"The majority of the dataset (97.8%) falls under the 'No Movement' category, indicating a lack of significant movement. On the other hand, the 'High Movement' category represents only a small fraction (1.1%) of the dataset. Additionally, the 'Moderate Movement' category comprises 0.7% of the samples, while the 'Low Movement' category accounts for 0.4% of the dataset."

In the Oversampling section on page 6, we clarify the representation of all classes:

"All other classes, including "High Movement," "Moderate Movement," and "Low Movement," represent minority classes, each constituting only 1%, 0.7%, and 0.4% of the total data, respectively."

Furthermore, on page 6, in the Oversampling section, we now explicitly state:

"By utilizing the characteristics of existing samples from the minority classes, we created new data points, thereby increasing the representation of the 'High Movement,' 'Moderate Movement,' and 'Low Movement' classes."

4. The main results are synthesised in Tables 5 and 6. In my opinion, these two tables are not enough to convey the effect of oversampling. In most cases the data without oversampling returns better scores than the oversampled data in all the metrics, not allowing the reader to understand the cause. Furthermore, scores so close to 1 might suggest a data leakage between training and model testing. It could be worth it to revise the data-splitting procedure and implement the pipeline with cross-validation to avoid this issue.

R: We appreciate your valuable feedback and have taken your comments into careful consideration. To address your concerns regarding the impact of oversampling, we have revised our methodology by incorporating a 5-fold cross-validation approach. This enhancement ensures a more robust evaluation of model performance, minimizing the risk of data leakage between the training and testing phases.

Upon implementing this cross-validation technique, we re-evaluated the results and observed a consistent improvement in the performance of models utilizing K-Means SMOTE for oversampling. The revised Tables 5 and 6 now accurately reflect the effectiveness of oversampling techniques, particularly highlighting the superiority of K-Means SMOTE in enhancing predictive accuracy.

We have updated the manuscript to include this important modification in the "Model Execution, Minimization, and Handling Class Imbalance" section on page 9, providing a clear description of the revised methodology.

“A rigorous process was followed to develop an effective model for predicting the intensity of soil movement. The dataset was partitioned into a 70:30 ratio, with 70% allocated for training and the remaining 30% for testing. To tackle the class imbalance issue in the training data, oversampling techniques were applied exclusively to the training set, ensuring a balanced representation of all three classes. The oversampling methods were not extended to the testing data, preserving its original distribution. Following the balancing process, a suite of ML models underwent training using a 5-fold cross-validation (5-CV) approach to the training data (Kumar et al., 2023). The models were optimized by employing grid search methodology, systematically exploring various parameter combinations that maximized the average cross-validation accuracy during training. The training performance, assessed through 5-CV, reflected the models' effectiveness with the optimized parameters. Subsequently, the models with the best parameters found during training were tested on the independent testing data, and their performance metrics were reported as indicative of their predictive capabilities. The evaluation primarily focused on accuracy metrics to determine how effectively the models predicted the intensity of soil movement.”

We have also added a new reference.

Kumar, P., Priyanka, P., Dhanya, J., Uday, K. V., & Dutt, V.: Analyzing the Performance of Univariate and Multivariate Machine Learning Models in Soil Movement Prediction: A Comparative Study. *IEEE Access*, 11, 62368–62381, 2023

Additionally, the results section on page 11 has been amended to present the latest findings obtained through 5-fold cross-validation.

“Table 5 presents the training results of different classification models combined with various oversampling techniques for landslide prediction. These results provide valuable insights into the performance of each model when trained on the training dataset with and without oversampling. The dynamic ensemble model with K-Mean SMOTE emerges as the best model in training, achieving outstanding accuracy, precision, recall, and F1 scores of 0.996, 0.996, 0.996, and 0.996, respectively. The dynamic ensemble model with SMOTE, Borderline SMOTE, and ADASYN techniques also showed similar performance with 0.995 F1 scores. It demonstrates remarkable predictive capability by achieving perfect accuracy in oversampling scenarios. When the model is trained without oversampling, its accuracy, precision, recall, and F1 score are notably lower, with values of 0.981, 0.557, 0.386, and 0.436, respectively.

Table 6 presents the test results of various classification models combined with different oversampling techniques for landslide prediction. Among them, the dynamic ensemble model utilizing the K-Mean SMOTE technique demonstrates exceptional performance in accurately predicting landslides on unseen data. It achieves impressive accuracy, precision, and recall rates of

0.994, 0.882, and 0.945, respectively, along with an F1 score of 0.911. These outstanding results confirm the effectiveness of the dynamic ensemble approach when combined with K-Mean SMOTE for accurate soil movement prediction. Notably, it is crucial to highlight the impact of oversampling on the performance of the dynamic ensemble model. When the model is tested without oversampling, its accuracy, precision, recall, and F1 score are notably lower, with values of 0.981, 0.557, 0.386, and 0.436, respectively. The best-performing model is highlighted in bold in Table 6.

Additionally, the dynamic ensemble model incorporating SMOTE emerges as the second-best model in the test phase, showcasing high accuracy, precision, and recall rates of 0.993, 0.872, and 0.950, respectively, along with an F1 score of 0.907. Moreover, it is noteworthy that K-Means SMOTE consistently outperformed other oversampling techniques across all models during the test performance evaluations, establishing itself as the optimal technique. In addition, the SMOTE technique consistently secured the second-best position across all models. This underscores the discernible effectiveness of K-Means SMOTE in generating oversampling for the soil movement dataset. The success of K-Means SMOTE can be attributed to its ability to identify clusters within the minority class and select similar features for oversampling. The IR employed by K-Means SMOTE aids in determining the appropriate degree of oversampling for the minority class, ensuring a balanced representation of classes in synthetic samples.

Moreover, the absence of oversampling techniques negatively impacted the models' performance in both training and testing. Without oversampling, the models exhibited lower accuracy, precision, recall, and F1 scores during training and testing, emphasizing the challenges posed by class imbalance. In the absence of balanced representation through oversampling, the models struggled to effectively learn and generalize from the imbalanced dataset. Consequently, this underscores the pivotal role of oversampling in mitigating class imbalance issues, leading to substantial enhancements in predictive accuracy and overall model robustness during both training and testing evaluations."

Table 5. The results of the ML models from the training dataset.

Model	Oversampling Technique	Accuracy	Precision	Recall	F1 Score
AdaBoost	SMOTE	0.632	0.638	0.632	0.632
	K-Means SMOTE	0.641	0.646	0.641	0.631
	Borderline SMOTE	0.663	0.670	0.663	0.659
	ADASYN	0.618	0.622	0.618	0.618
	Without Oversampling	0.980	0.556	0.357	0.393
XGBoost	SMOTE	0.921	0.921	0.921	0.921
	K-Means SMOTE	0.926	0.926	0.926	0.926
	Borderline SMOTE	0.973	0.973	0.973	0.973
	ADASYN	0.915	0.916	0.915	0.915
	Without Oversampling	0.994	0.983	0.814	0.882
Light GBM	SMOTE	0.920	0.920	0.920	0.920
	K-Means SMOTE	0.939	0.940	0.939	0.939
	Borderline SMOTE	0.963	0.963	0.963	0.963
	ADASYN	0.915	0.916	0.915	0.915
	Without Oversampling	0.991	0.845	0.791	0.807
CatBoost	SMOTE	0.860	0.860	0.860	0.859

	K-Means SMOTE	0.876	0.876	0.876	0.876
	Borderline SMOTE	0.932	0.932	0.932	0.932
	ADASYN	0.859	0.859	0.859	0.859
	Without Oversampling	0.983	0.797	0.399	0.469
RF	SMOTE	0.731	0.742	0.731	0.728
	K-Means SMOTE	0.734	0.748	0.734	0.729
	Borderline SMOTE	0.795	0.806	0.795	0.797
	ADASYN	0.732	0.747	0.732	0.728
	Without Oversampling	0.982	0.905	0.325	0.372
MLP	SMOTE	0.902	0.903	0.902	0.901
	K-Means SMOTE	0.944	0.945	0.944	0.944
	Borderline SMOTE	0.961	0.962	0.961	0.962
	ADASYN	0.942	0.943	0.942	0.942
	Without Oversampling	0.979	0.635	0.309	0.339
LSTM	SMOTE	0.747	0.750	0.747	0.745
	K-Means SMOTE	0.767	0.769	0.767	0.766
	Borderline SMOTE	0.779	0.781	0.779	0.778
	ADASYN	0.756	0.759	0.756	0.755
	Without Oversampling	0.758	0.760	0.758	0.756
Dynamic Ensemble	SMOTE	0.995	0.995	0.995	0.995
	K-Means SMOTE	**0.996**	**0.996**	**0.996**	**0.996**
	Borderline SMOTE	0.995	0.995	0.995	0.995
	ADASYN	0.995	0.995	0.995	0.995
	Without Oversampling	0.981	0.557	0.386	0.436

Table 6. The results of the ML models from the test dataset.

Model	Oversampling Technique	Accuracy	Precision	Recall	F1 Score
AdaBoost	SMOTE	0.798	0.301	0.646	0.313
	K-Means SMOTE	0.865	0.313	0.610	0.342
	Borderline SMOTE	0.804	0.313	0.598	0.326
	ADASYN	0.788	0.293	0.625	0.300
	Without Oversampling	0.979	0.419	0.313	0.340
XGBoost	SMOTE	0.957	0.509	0.872	0.610
	K-Means SMOTE	0.957	0.486	0.807	0.576
	Borderline SMOTE	0.954	0.480	0.770	0.560
	ADASYN	0.958	0.517	0.876	0.619
	Without Oversampling	0.981	0.618	0.402	0.461
Light GBM	SMOTE	0.951	0.495	0.858	0.591
	K-Means SMOTE	0.952	0.475	0.796	0.561
	Borderline SMOTE	0.954	0.474	0.735	0.548
	ADASYN	0.951	0.488	0.865	0.586
	Without Oversampling	0.978	0.511	0.447	0.467
CatBoost	SMOTE	0.944	0.439	0.831	0.524
	K-Means SMOTE	0.945	0.443	0.793	0.528
	Borderline SMOTE	0.948	0.431	0.747	0.510
	ADASYN	0.948	0.433	0.791	0.517
	Without Oversampling	0.980	0.664	0.389	0.442
RF	SMOTE	0.829	0.342	0.737	0.367
	K-Means SMOTE	0.831	0.336	0.687	0.355
	Borderline SMOTE	0.833	0.321	0.632	0.346
	ADASYN	0.828	0.350	0.689	0.364
	Without Oversampling	0.978	0.477	0.268	0.280
MLP	SMOTE	0.887	0.414	0.945	0.498
	K-Means SMOTE	0.928	0.499	0.959	0.602
	Borderline SMOTE	0.907	0.423	0.742	0.473

	ADASYN	0.909	0.454	0.942	0.547
	Without Oversampling	0.978	0.635	0.308	0.339
LSTM	SMOTE	0.856	0.318	0.684	0.352
	K-Means SMOTE	0.940	0.402	0.736	0.473
	Borderline SMOTE	0.925	0.384	0.720	0.448
	ADASYN	0.887	0.326	0.556	0.361
	Without Oversampling	0.827	0.312	0.710	0.339
Dynamic Ensemble	SMOTE	0.993	0.872	0.950	0.907
	K-Means SMOTE	**0.994**	**0.882**	**0.945**	**0.911**
	Borderline SMOTE	0.993	0.900	0.869	0.880
	ADASYN	0.993	0.854	0.952	0.898
	Without Oversampling	0.982	0.695	0.434	0.506

5. Chapter 7 is just conclusions; the critical investigation of results (i.e., the discussion) is completely missing.

R: We sincerely appreciate your thorough review of Chapter 7. Your insightful comments have guided us in making important revisions to ensure the completeness of the document. We have now addressed this concern by incorporating a comprehensive discussion section on page 13, covering critical investigation, outcomes of the experiment, implications of oversampling techniques, limitations, and key findings.

The Discussion and Conclusion Section is revised as follows:

In summary, the threat posed by landslides requires the development of effective prediction frameworks, although modeling the chaotic nature of natural data remains challenging. The analyzed dataset exhibited a significant class imbalance, with the majority class dominating the samples. This distribution imbalance necessitated careful consideration and appropriate techniques to address the issue.

Various oversampling techniques, including SMOTE and its extensions (K-Means SMOTE, Borderline SMOTE, and ADASYN), were employed to tackle the class imbalance. ADASYN, which focuses on the minority class boundary, effectively generated synthetic data points and improved the class distribution balance.

Multiple classification models, such as ADABoost, XGBoost, Light GBM, CatBoost, RF, MLP, LSTM, and a dynamic ensemble, were evaluated to predict soil movement. The grid search approach and 5-CV were employed to optimize the hyperparameters of each model. The training results highlight the significant impact of oversampling on model performance. The dynamic ensemble model, particularly when coupled with K-Means SMOTE, emerges as the standout performer in the training phase. Achieving remarkable accuracy, precision, recall, and F1 scores of 0.996, 0.996, 0.996, and 0.996, respectively, this model demonstrates superior predictive capabilities.

Furthermore, these models were tested to assess their ability to generalize well to unseen data. The testing results showcased the dynamic ensemble model with K-Means SMOTE as the top performer, achieving an outstanding accuracy of 0.994, precision of 0.882, recall of 0.945, and an

F1 score of 0.911. This confirms that the exceptional performance observed in training extends to the testing phase, emphasizing the robustness and reliability of the dynamic ensemble approach with K-Means SMOTE. Moreover, the dynamic ensemble model incorporating SMOTE emerges as the second-best model in the test phase, showcasing high accuracy, precision, and recall rates of 0.993, 0.872, and 0.950, respectively, along with an F1 score of 0.907. This result reinforces the reliability and robustness of the model in tackling landslide prediction tasks.

Furthermore, the dynamic ensemble model incorporating SMOTE emerges as the second-best model in the test phase, showcasing high accuracy, precision, and recall rates of 0.993, 0.872, and 0.950, respectively, along with an F1 score of 0.907. This result reinforces the reliability and robustness of the model in tackling landslide prediction tasks.

The superior performance of the K-Means SMOTE technique can be attributed to its ability to identify clusters within the minority class and generate synthetic samples that maintain the underlying structure of the data. By considering the IR, K-Means SMOTE ensures a balanced representation of classes in the synthetic samples, contributing to improved model generalization and predictive accuracy. Furthermore, the lack of oversampling adversely affected both training and testing performances. The models faced challenges in learning and generalizing from the imbalanced dataset without a balanced representation.

On the other hand, the success of the dynamic ensemble model, comprising AdaBoost, XGBoost, Light GBM, CatBoost, and Random Forest, can be attributed to the complementary strengths of these diverse algorithms. Ensemble methods leverage the collective decision-making power of multiple models, each capturing different aspects of the underlying data patterns. The combination of boosting algorithms like AdaBoost, gradient boosting methods like XGBoost, tree-based models like Light GBM and CatBoost, and the robustness of RF creates a robust and versatile ensemble that excels in handling various aspects of the dataset, contributing to its overall superior performance.

In summary, the findings underscore the critical role of oversampling techniques, especially K-Means SMOTE, in enhancing the predictive performance of landslide prediction models. The success of the dynamic ensemble model further highlights the importance of ensemble techniques in aggregating diverse model predictions for improved accuracy.

Despite these achievements, it is crucial to acknowledge the study's limitations. The generalizability of the findings to different geological conditions or regions may be restricted due to the specificity of the dataset. The synthetic data points generated through oversampling, while effective, may only capture part of the complexity inherent in real-world landslide occurrences. The choice of classification models and hyperparameter settings introduces a level of bias, with alternative configurations potentially yielding different results. Additionally, relying on historical data may limit the model's ability to account for future changes or unforeseen events, such as changes in rainfall intensity, seismic activity, or human influences.
In future work, the exploration of encoder-decoder models or transformer models on the class-imbalanced movement dataset is planned. These models, known for their success in sequence-to-

sequence tasks, may offer improvements in classification accuracy and address class imbalance challenges. This avenue of experimentation aims to provide valuable insights into the suitability of advanced models for analyzing and modeling imbalanced movement data.

To sum up, the study contributes to the understanding of landslide risks and supports the development of effective preventive measures. The combination of robust oversampling techniques, ensemble modeling, and a systematic approach to hyperparameter tuning yields a promising framework for accurate landslide prediction. The work presented lays the groundwork for future research aimed at refining models and addressing the inherent challenges in landslide prediction tasks.

Dear Anonymous Referee #2,

Thank you for your positive and constructive comments. We have carefully considered your comments and made several revisions to the manuscript (marked Response: in blue font). Firstly, we conducted a parameter variation analysis on different datasets to assess how parameters change across datasets. Secondly, we refined the results by incorporating insights from 5-fold cross-validation. Lastly, we enhanced the discussion and conclusion sections to provide a clearer understanding of our findings. We believe that these revisions will significantly improve the clarity and impact of our work.

Kind Regards,
Praveen Kumar

Anonymous Referee #2 Comments

This paper presents the development of machine learning (ML) models with oversampling techniques to address the class imbalance issue, essential to developing a robust soil movement prediction system.

The paper is well-written and easy to follow. I have some significant questions regarding the proposed methods:

Response #0: Thank you for your kind words regarding the clarity and readability of our paper. We appreciate your feedback and welcome your questions regarding the proposed methods. We addressed your valuable comments comprehensively as follows:

1. How much does the model parameters value change with different training data sets? Also, authors should take different training sets for their method evaluation.

Response #1: Thank you for your valuable feedback. We have incorporated your kind suggestion by utilizing a 5-training datasets method to evaluate our machine-learning model with different parameter ranges. In this 5-training datasets (5-TD) method, our training dataset was split into 5 independent datasets, and the machine learning model's parameters were optimized on each of these individual sets. The parameter analysis, which examines how the model parameters' values changed with different training datasets, is discussed in detail in the parameter analysis subsection on page 10 of the manuscript. This analysis provides insights into the mean and standard deviation of the parameter values across the different TDs, shedding light on the variability and consistency of model parameterization.

We have now introduced a new section titled "Model Execution, Minimization, and Handling Class Imbalance" on page 10 to provide further insights into the 5-TD and the 5-CV methods. We also expanded upon the parameter variance analysis result and optimized parameter section on page 11 and included a Train-Test Results section on page 12 to discuss the revised results comprehensively. We believe these enhancements strengthen the rigour and clarity of our methodology and results.

The different sections of the manuscript are summarized below.

5. Model Execution, Minimization, and Handling Class Imbalance

A rigorous process was followed to develop an effective model for predicting the intensity of soil movement. The dataset was partitioned into a 70:30 ratio, with 70% allocated for training and 30% for testing. To tackle the class imbalance issue in the training data, oversampling techniques were applied exclusively to the training set, ensuring a balanced representation of all three classes. The oversampling methods were not extended to the testing data, preserving its original distribution. In this study, we developed two methods, referred to as method 5-TD and method 5-CV. Method 5-TD was employed for parameter variation analysis across different datasets. On the other hand, method 5-CV was utilized for conducting 5-fold cross-validation (5-CV) to analyze the performance of the ML models.

5.1. Method 5-TD:

For method 5-TD, the training dataset was split into five training datasets, each utilized for parameter variation analysis. This involved training and optimizing the ML model on each dataset independently using the grid search method. Since each dataset possessed different optimal parameters, we calculated the mean and standard deviation (stdev) of the ML-optimized parameter values across all datasets to assess parameter variability. This enabled us to observe parameter variations across the ML models, providing insights into the sensitivity of the models to different dataset characteristics and parameter configurations. A lower stdev implied that the model maintained consistency across each dataset and demonstrated robust generalization capabilities. Conversely, a higher stdev suggested that the model encountered difficulties maintaining consistency across datasets, potentially hindering its ability to learn general patterns effectively. The evaluation primarily focused on F1 score metrics to determine how effectively the models predicted the intensity of soil movements in each of the 5 datasets.

5.2. Method 5-CV:

For method 5-CV, a suite of ML models underwent training using a 5-fold cross-validation approach (Kumar et al., 2023). In the 5-CV method, the training data was split into 5 datasets, where each dataset was alternately used for validation while the others were used for training. The models were optimized by employing grid search methodology and optimized based on performance on the 5 validation sets, and a single set of best-performing parameters was selected for each model. Subsequently, the models with the best parameters found during training were tested on the independent testing data, and their performance metrics were reported as indicative of their predictive capabilities. The evaluation primarily focused on F1 score metrics to determine how effectively the models predicted the intensity of soil movement across the 5 validation sets and the test set.

6. Results

6.1. Parameter Variation Analysis Result

Upon scrutinizing the parameter analysis presented in Table 4 from method 5-TD, a discernible trend emerged: models trained with oversampling techniques exhibit notably smaller stdevs than their counterparts trained without oversampling. For instance, when examining the AdaBoost model, we observe that the stdev of the number of trees parameter was 0 for the oversampling case. In contrast, it stood at 16.43 for the dataset without oversampling. This phenomenon underscores the stabilizing effect of oversampling on parameter estimates, mitigating the variability that may arise from imbalanced datasets.

Similarly, in the case of the RF model, the stdev of the number of trees parameter was 0 with oversampling, indicating consistent parameter values across folds. Conversely, for the dataset without oversampling, the stdev increased to 21.21, suggesting greater variability in parameter estimates. This trend

persisted across various models and parameters, highlighting the robustness imparted by oversampling techniques in stabilizing model performance.

Overall, these examples underscore the importance of oversampling in reducing parameter variability and ensuring consistent model behaviour, particularly in scenarios involving imbalanced datasets.

Table 4. The result of parameter variation analysis across five datasets from method 5-TD.

Model	Parameter	With Oversampling		Without Oversampling	
		Mean	stdev	Mean	stdev
AdaBoost	Number of Trees	80	0	62	16.43
	Learning Rate	0.66	0.22	0.9	0
XGBoost	Number of Trees	50	0	50	0
	Maximum Depth	20	0	10	0
	Learning Rate	0.5	0	0.68	0.16
Light GBM	Number of Trees	50	0	50	0
	Maximum Depth	20	0	20	0
	Learning Rate	0.5	0	0.6	0.12
CatBoost	Number of Trees	50	0	50	0
	Maximum Depth	20	0	20	0
	Learning Rate	0.8	0	0.66	0.13
RF	Number of Trees	80	0	50	21.21
	Maximum Depth	20	0	20	0
MLP	Look-back Period	2.8	0.44	3.6	1.34
	Layers	2	0	2	0
	Nodes in First Layer	130	67.08	130	67.08
	Nodes in Second Layer	200	0	60	54.77
	Learning Rate	0.78	0.16	0.64	0.28
LSTM	Look-back Period	4.6	0.89	4	1.41
	Layers	2	0	2	0
	Nodes in First Layer	90	22.36	70	27.39
	Nodes in Second Layer	160	54.77	100	61.24
	Learning Rate	0.84	0.08	0.86	0.05

6.2 Optimized Parameters

In method 5-CV, we optimized the parameters separately for the ML models using a 5-fold cross-validation process on the full training dataset. Table 5 presents each model's optimized parameter values obtained through the grid search in 5-CV on the training dataset. These parameters were carefully fine-tuned to ensure the best fit for the given data. In the case of AdaBoost, the optimized values included 80 trees and a learning rate of 0.6. The optimized values for the XGBoost model consisted of 50 trees, a learning rate of 0.3, and a maximum depth of 10. These settings were determined to enhance the model's performance in terms of both speed and accuracy.

Similarly, the Light GBM model underwent parameter optimization, selecting 50 trees, a learning rate of 0.5, and a maximum depth of 20. Next, the CatBoost model was also optimized, leading to entropy selection as the loss function, a learning rate of 0.8, 50 trees, and a maximum depth of 20. In the RF model, the optimized values were 80 for the number of trees and 20 for the maximum depth, and the evaluation criteria were set to "Gini." Likewise, the MLP model optimized its parameters with a look-back period of 3, 2 layers, and 200 nodes per layer. Similarly, the LSTM model consists of two layers with 100 and 200 nodes in the first and second layers and utilizes a ReLU activation function. Lastly, the dynamic ensemble model in this study incorporated the optimized RF, CatBoost, XGBoost, Light GBM, and AdaBoost models to improve the accuracy of landslide analysis predictions. By leveraging the strengths of these individually optimized models, as mentioned above, the dynamic ensembling model aimed to improve the accuracy and reliability of landslide analysis predictions.

Table 5. The best value of the parameters was calibrated from the training data using method 5-CV.

Model	Parameter	Best Value of Parameter
AdaBoost	Number of Trees	80
	Learning Rate	0.6
XGBoost	Number of Trees	50
	Learning Rate	0.3
	Maximum Depth	10
Light GBM	Number of Trees	50
	Learning Rate	0.5
	Maximum Depth	20
CatBoost	Loss Function	Entropy
	Learning Rate	0.8
	Number of Trees	50
	Maximum Depth	20
RF	Number of Trees	80
	Criteria	Gini

	Maximum Depth	20
MLP	Look-back Period	3
	Layers	2
	Nodes Per Layer	200 in both layers
	Learning Rate	0.6
LSTM	Look-back Period	5
	LSTM Units	100 in the first and 200 in the second layer
	Activation Function	ReLU
	Learning Rate	0.9

6.3. Train-Test Results

Table 6 presents the training results of different classification models evaluated using 5-fold cross-validation on the training dataset and various oversampling techniques for landslide prediction, utilizing method 5-CV. In Table 6, C0, C1, C2, and C3 represent no movement, low movement, moderate movement, and high movement classes' accuracies, respectively. These results provide valuable insights into the performance of each model when trained on the training dataset with and without oversampling. The XGBoost model with K-Mean SMOTE emerged as the best model in training, achieving outstanding accuracy, precision, recall, and F1 scores of 0.999, 0.999, 0.999, and 0.999, respectively. The dynamic ensemble model with K-Mean SMOTE and Borderline SMOTE techniques also performed similarly with 0.998 F1 scores. It demonstrates remarkable predictive capability by achieving perfect accuracy in oversampling scenarios. When the XGBoost model was trained without oversampling, its accuracy, precision, recall, and F1 score were notably lower, with values of 0.999, 0.999, 0.971, and 0.985, respectively.

Table 7 presents the test results of various classification models combined with different oversampling techniques for landslide prediction (here, models were trained using the method 5-CV). In Table 7, C0, C1, C2, and C3 represent no movement, low movement, moderate movement, and high movement classes' accuracies, respectively. Among them, the dynamic ensemble model utilizing the K-Mean SMOTE technique demonstrated exceptional performance in accurately predicting landslides on unseen data. It achieves impressive accuracy, precision, and recall rates of 0.995, 0.995, and 0.995, respectively, along with an F1 score of 0.95. These outstanding results confirm the effectiveness of the dynamic ensemble approach when combined with K-Mean SMOTE for accurate soil movement prediction. Similarly, the Borderline SMOTE technique also showed similar performance with accuracy, precision, recall, and an F1 score of 0.995 for all. When the model is tested without oversampling, its accuracy, precision, recall, and F1 score are notably lower, with values of 0.981, 0.646, 0.397, and 0.462, respectively. The best-performing model is highlighted in bold in Table 6 and Table 7.

Moreover, it is noteworthy that K-Means SMOTE consistently outperformed other oversampling techniques across all models during the test performance evaluations, establishing itself as the optimal technique. Notably, it is crucial to highlight the impact of oversampling on the performance of the dynamic ensemble model. This underscores the discernible effectiveness of K-Means SMOTE in generating

oversampling for the soil movement dataset. The success of K-Means SMOTE can be attributed to its ability to identify clusters within the minority class and select similar features for oversampling. The IR employed by K-Means SMOTE aids in determining the appropriate degree of oversampling for the minority class, ensuring a balanced representation of classes in synthetic samples.

Moreover, the absence of oversampling techniques negatively impacted the models' performance in both training and testing. Without oversampling, the models exhibited lower accuracy, precision, recall, and F1 scores during training and testing, emphasizing the challenges posed by class imbalance. In the absence of balanced representation through oversampling, the models struggled to effectively learn and generalize from the imbalanced dataset. Consequently, this underscores the pivotal role of oversampling in mitigating class imbalance issues, leading to substantial enhancements in predictive accuracy and overall model robustness during training and testing evaluations.

Models trained with oversampling techniques consistently demonstrate comparable performance across both training and testing datasets, indicating a lack of overfitting. Conversely, models trained without oversampling, notably RF, MLP, LSTM, and Dynamic Ensemble, exhibit signs of overfitting, as evidenced by significantly higher performance metrics on the training dataset relative to the testing dataset. This observation underscores the effectiveness of oversampling techniques in mitigating overfitting by enhancing the model's ability to generalize to unseen data.

Comparing the dynamic ensemble model with other classification models, it becomes evident that the dynamic ensemble model with K-Mean SMOTE consistently outperformed the rest, highlighting their effectiveness in accurately predicting landslides.

These findings underscore the importance of carefully selecting appropriate ML models and employing suitable oversampling techniques to address the class imbalance challenge in soil movement prediction. They provide valuable insights into the performance and suitability of these models and techniques for enhancing landslide prediction accuracy, ultimately enabling proactive measures to mitigate landslide risks.

Table 6. Results of ML models obtained from the training dataset using 5-fold cross-validation in method 5-CV.

Model	Oversampling Technique	Accuracy					Precision	Recall	F1 Score
		C0	C1	C2	C3	Overall			
AdaBoost	SMOTE	0.942	0.562	0.640	0.817	0.747	0.748	0.747	0.747
	K-Means SMOTE	0.948	0.760	0.675	0.855	0.807	0.809	0.807	0.806
	Borderline SMOTE	0.919	0.565	0.667	0.815	0.740	0.741	0.740	0.740
	ADASYN	0.934	0.552	0.649	0.798	0.740	0.741	0.740	0.740
	Without Oversampling	0.995	0.250	0.243	0.341	0.980	0.575	0.465	0.506
XGBoost	SMOTE	0.995	0.999	0.999	0.997	0.998	0.998	0.998	0.998
	K-Means SMOTE	0.997	0.999	0.999	0.998	0.999	0.999	0.999	0.999
	Borderline SMOTE	0.996	0.999	0.999	0.998	0.998	0.998	0.998	0.998
	ADASYN	0.994	0.999	0.999	0.997	0.998	0.998	0.998	0.998
	Without Oversampling	1.000	0.995	0.953	0.906	0.999	0.999	0.971	0.985
Light GBM	SMOTE	0.984	0.994	0.999	0.988	0.991	0.991	0.991	0.991

	K-Means SMOTE	0.991	0.998	0.998	0.996	0.996	0.996	0.996	0.996
	Borderline SMOTE	0.985	0.999	0.999	0.995	0.995	0.995	0.995	0.995
	ADASYN	0.983	0.994	0.998	0.987	0.991	0.991	0.991	0.991
	Without Oversampling	1.000	1.000	1.000	0.976	0.994	0.999	0.999	0.996
CatBoost	SMOTE	0.990	0.999	0.999	0.997	0.997	0.997	0.997	0.997
	K-Means SMOTE	0.991	0.999	0.999	0.997	0.997	0.997	0.997	0.997
	Borderline SMOTE	0.992	0.999	0.999	0.997	0.997	0.997	0.997	0.997
	ADASYN	0.991	0.999	0.999	0.997	0.996	0.996	0.996	0.996
	Without Oversampling	0.999	0.924	0.916	0.735	0.997	0.997	0.903	0.946
RF	SMOTE	0.920	0.892	0.951	0.905	0.921	0.923	0.921	0.922
	K-Means SMOTE	0.920	0.921	0.959	0.902	0.925	0.928	0.925	0.926
	Borderline SMOTE	0.948	0.969	0.988	0.959	0.967	0.967	0.967	0.967
	ADASYN	0.921	0.898	0.945	0.899	0.915	0.917	0.915	0.915
	Without Oversampling	1.000	0.701	0.682	0.537	0.992	0.995	0.742	0.841
MLP	SMOTE	0.959	0.976	0.997	0.952	0.961	0.961	0.961	0.961
	K-Means SMOTE	0.940	0.996	0.984	0.957	0.974	0.974	0.974	0.974
	Borderline SMOTE	0.968	0.974	0.989	0.913	0.964	0.964	0.964	0.964
	ADASYN	0.929	0.975	0.981	0.984	0.961	0.961	0.961	0.961
	Without Oversampling	0.997	0.016	0.000	0.056	0.980	0.693	0.336	0.381
LSTM	SMOTE	0.882	0.841	0.881	0.896	0.875	0.884	0.875	0.877
	K-Means SMOTE	0.980	0.996	0.992	0.968	0.984	0.984	0.984	0.984
	Borderline SMOTE	0.946	0.954	0.997	0.965	0.966	0.966	0.966	0.966
	ADASYN	0.955	0.979	0.997	0.955	0.971	0.971	0.971	0.971
	Without Oversampling	0.999	0.859	0.925	0.700	0.995	0.979	0.871	0.919
Dynamic Ensemble	SMOTE	0.992	0.999	0.999	0.999	0.997	0.997	0.997	0.997
	K-Means SMOTE	0.994	0.999	0.999	0.999	0.998	0.998	0.998	0.998
	Borderline SMOTE	0.997	0.999	0.999	0.998	0.998	0.998	0.998	0.998
	ADASYN	0.992	0.999	0.999	0.998	0.997	0.997	0.997	0.997

	Without Oversampling	1.000	0.951	0.944	0.770	0.997	0.999	0.916	0.954

Table 7. Results of ML models obtained from the testing dataset in method 5-CV.

Model	Oversampling Technique	Accuracy					Precision	Recall	F1 Score
		C0	C1	C2	C3	Overall			
AdaBoost	SMOTE	0.939	0.548	0.436	0.763	0.932	0.383	0.671	0.442
	K-Means SMOTE	0.946	0.583	0.436	0.681	0.939	0.382	0.662	0.445
	Borderline SMOTE	0.917	0.595	0.462	0.756	0.911	0.374	0.682	0.423
	ADASYN	0.995	0.226	0.205	0.230	0.978	0.514	0.414	0.447
	Without Oversampling	0.931	0.524	0.436	0.681	0.924	0.360	0.643	0.412
XGBoost	SMOTE	0.991	0.976	0.974	0.837	0.989	0.774	0.945	0.846
	K-Means SMOTE	0.993	0.952	0.949	0.785	0.990	0.787	0.920	0.842
	Borderline SMOTE	0.994	0.905	0.769	0.733	0.990	0.803	0.850	0.823
	ADASYN	0.990	0.988	0.974	0.830	0.988	0.761	0.946	0.837
	Without Oversampling	0.996	0.250	0.026	0.333	0.980	0.553	0.401	0.447
Light GBM	SMOTE	0.983	0.905	0.974	0.748	0.980	0.656	0.903	0.750
	K-Means SMOTE	0.984	0.917	0.872	0.704	0.980	0.654	0.869	0.737
	Borderline SMOTE	0.990	0.738	0.667	0.637	0.983	0.695	0.758	0.720
	ADASYN	0.981	0.917	0.974	0.741	0.978	0.638	0.903	0.735
	Without Oversampling	0.996	0.214	0.205	0.326	0.980	0.547	0.435	0.472
CatBoost	SMOTE	0.986	0.964	0.974	0.852	0.984	0.705	0.944	0.799
	K-Means SMOTE	0.988	0.952	0.974	0.815	0.986	0.726	0.932	0.810
	Borderline SMOTE	0.990	0.798	0.641	0.689	0.984	0.720	0.779	0.743
	ADASYN	0.987	0.988	0.974	0.859	0.985	0.722	0.952	0.814
	Without Oversampling	0.997	0.226	0.179	0.311	0.981	0.611	0.428	0.487
RF	SMOTE	0.988	0.988	0.974	0.970	0.988	0.763	0.980	0.851
	K-Means SMOTE	0.995	0.917	0.821	0.867	0.993	0.885	0.900	0.889
	Borderline SMOTE	0.991	0.976	0.974	0.956	0.991	0.801	0.974	0.875
	ADASYN	0.989	0.988	0.974	0.978	0.988	0.757	0.982	0.848

	Without Oversampling	0.998	0.190	0.051	0.289	0.980	0.676	0.382	0.440
MLP	SMOTE	0.958	1.000	1.000	0.948	0.958	0.554	0.977	0.671
	K-Means SMOTE	0.965	0.988	0.974	0.830	0.964	0.578	0.939	0.689
	Borderline SMOTE	0.937	0.750	0.641	0.659	0.932	0.444	0.747	0.518
	ADASYN	0.927	1.000	0.974	0.963	0.928	0.554	0.966	0.652
	Without Oversampling	0.995	0.012	0.026	0.015	0.974	0.380	0.262	0.270
LSTM	SMOTE	0.878	0.774	0.897	0.815	0.877	0.451	0.841	0.522
	K-Means SMOTE	0.981	0.869	0.923	0.763	0.977	0.693	0.884	0.766
	Borderline SMOTE	0.948	0.917	1.000	0.919	0.948	0.527	0.946	0.636
	ADASYN	0.953	0.952	1.000	0.911	0.953	0.552	0.954	0.661
	Without Oversampling	0.996	0.488	0.667	0.415	0.985	0.804	0.642	0.704
Dynamic Ensemble	SMOTE	0.978	0.999	0.999	0.997	0.994	0.994	0.994	0.994
	K-Means SMOTE	0.988	0.998	0.998	0.996	0.995	0.995	0.995	0.995
	Borderline SMOTE	0.982	0.999	0.999	0.997	0.995	0.995	0.995	0.995
	ADASYN	0.979	0.999	0.999	0.997	0.994	0.994	0.994	0.994
	Without Oversampling	0.998	0.167	0.128	0.296	0.981	0.646	0.397	0.462

We have also added a new reference.

Kumar, P., Priyanka, P., Dhanya, J., Uday, K. V., & Dutt, V.: Analyzing the Performance of Univariate and Multivariate Machine Learning Models in Soil Movement Prediction: A Comparative Study. IEEE Access, 11, 62368–62381, 2023

2. The results should also contain the accuracy of each class. A truth table will be beneficial to understanding the method's performance.

Response #2: Thank you for your valuable suggestion. We have incorporated your kind comment by including the accuracy of each class in the training and testing results of method 5-CV in Tables 6 and 7, respectively. Additionally, we have provided the confusion matrix of the training and testing of the best-performing model, Dynamic Ensemble, with the K-Mean SMOTE technique as Figure 3 on page number 17. This allows for a more comprehensive understanding of the method's performance, particularly in terms of class-wise accuracy.

Figure 3 illustrates the confusion matrix depicting the performance of the Dynamic Ensemble model on both the training and testing datasets, utilizing the K-Mean SMOTE oversampling technique. The confusion

matrix provides a comprehensive overview of the model's classification accuracy by presenting the true and predicted labels across different classes. The Dynamic Ensemble model demonstrates robust performance in the training dataset, as evidenced by the high counts along the diagonal, indicating a substantial number of correct predictions across all classes. Similarly, in the testing dataset, the model maintains its efficacy, with the majority of samples correctly classified across various classes.

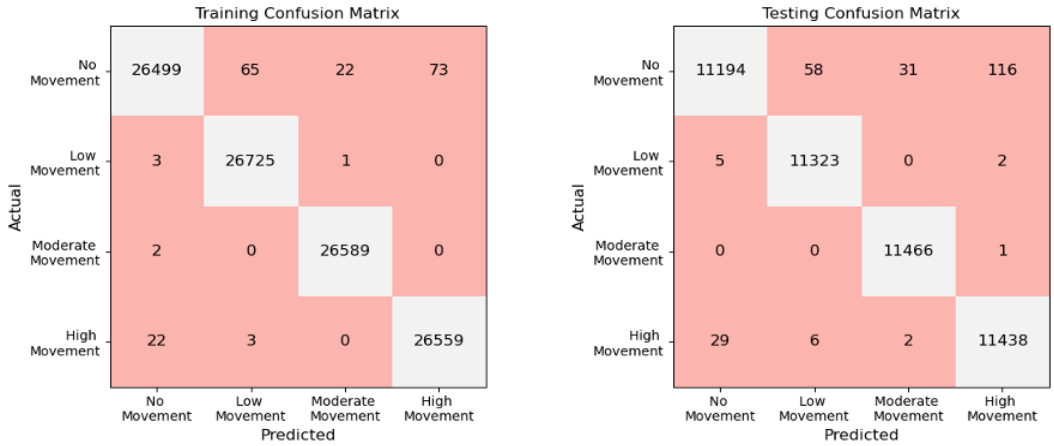

Figure 3. Confusion matrix depicting the performance of the Dynamic Ensemble model on the training and testing datasets using the K-Mean SMOTE oversampling technique and 5-CV.

3. The RF model has 100 % performance for training that might be overfitting in the model. Please check the overfitting in the model

Response #3: Thank you for your feedback. We are pleased to inform you that after revising our evaluation method 5-TD and implementing the 5-fold cross-validation technique in method 5-CV, we have observed significant improvements in the performance of the RF model. Specifically, the RF model no longer exhibited 100% accuracy, indicating a reduction in the overfitting problem that was previously observed. By incorporating 5-CV, we were able to enhance the robustness of our evaluation process and obtain more reliable performance estimates for the RF model. We appreciate your insightful suggestion, which has contributed to the refinement of our methodology and the improvement of our model's performance.

4. The paper should include the precautions authors took to ensure no information from training samples was mixed with the testing samples.

Response #4: Thank you for your comment. In our revised evaluation method 5-CV, we took specific precautions to ensure that no information from the training samples was mixed with the testing samples. We adopted a rigorous approach where we split our dataset into 70% for training and 30% for testing. Within the 70% training dataset, we implemented 5-CV and optimized the parameters of our models. These optimized models were then exclusively tested on the remaining 30% testing dataset. Additionally, we divided the 70% training dataset into five subsets and conducted parameter variation analysis of ML models on these individual subsets. By following this methodology, we ensured that there was no mixing of information between the training and testing datasets, thus maintaining the integrity of our evaluation process. Also, similar care was taken in the 5-TD method, where parameters were obtained separately across the 5 datasets with no mixing.

5. It will be good to compare results without a balanced dataset versus a balanced dataset (using oversampling techniques) in a plot. Also, discuss the reasons why no oversampling performs well over oversampling in some cases.

Response #5: Thank you for your valuable suggestion. We have incorporated a comparison of results between a balanced dataset (achieved through oversampling techniques) and an imbalanced dataset (without oversampling) in Figure 2 on page 17 of the revised manuscript. This comparison contrasts the performance of models trained on synthetic data generated by the best-performing K-Mean SMOTE technique with those trained without oversampling techniques across all machine learning models. Additionally, in the discussion and conclusion section, we elaborate on why oversampling techniques often lead to improved results. We highlight that oversampling methods are crucial in addressing class imbalances by providing the model with more representative training data.

In Figure 2, we juxtaposed the performance metrics obtained using K-Means SMOTE against those obtained without oversampling across various machine learning models. In Figure 2, the blue bars represent the F1 score achieved with K-Means SMOTE (oversampling), while the orange bars represent the F1 score without oversampling. Notably, when comparing the performance in the test dataset using the F1 score metric, the oversampling dataset generated with K-Means SMOTE consistently yielded superior results compared to the without oversampling approach. For instance, in the case of the AdaBoost model, K-Means SMOTE resulted in an F1 score of 0.412 for the without oversampling technique, whereas it achieved an F1 score of 0.445 for K-Means SMOTE. Similarly, in the XGBoost model, the F1 score improved from 0.447 without oversampling to 0.842 with K-Means SMOTE. This trend persisted across various other models such as Light GBM, CatBoost, RF, MLP, LSTM, and Dynamic Ensemble, where K-Means SMOTE consistently demonstrated superior performance in terms of F1 score compared to without oversampling. These results underscore the effectiveness of K-Means SMOTE in enhancing the predictive performance of ML models for soil movement prediction tasks.

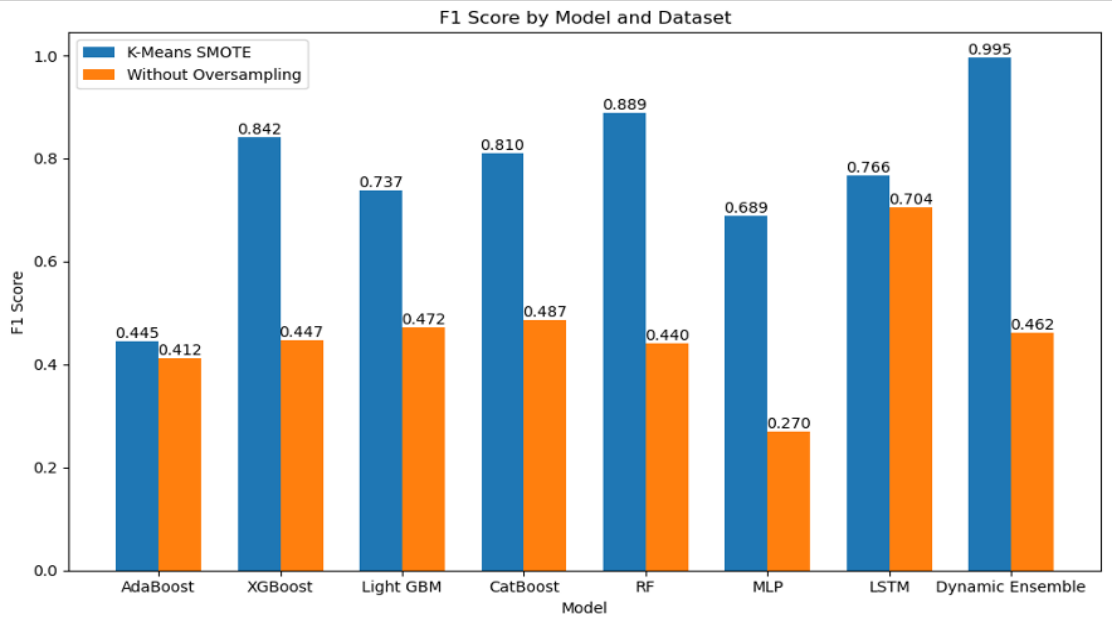

Figure 2. Comparison of F1 Score performance between K-Means SMOTE and without oversampling techniques across various ML models for soil movement prediction in testing. Blue bars represent F1 scores achieved with K-Means SMOTE, while orange bars represent F1 scores obtained without oversampling.

And also discuss the reasons why no oversampling performs well over oversampling in the discussion and conclusion section on the page number 18 as following:

The superior performance demonstrated by oversampling techniques compared to without oversampling can be attributed to several factors. Firstly, oversampling techniques address class imbalance by generating synthetic samples for minority classes, thus providing the model with more representative training data. This allows the ML model to learn the underlying patterns of the minority class more effectively, leading to improved classification performance. Additionally, oversampling techniques help reduce the risk of overfitting by providing a more balanced representation of the dataset, enhancing the model's ability to generalize to unseen data. Moreover, by increasing the diversity of the training data, oversampling techniques enable the model to capture a wider range of variation within the dataset, resulting in better generalization performance. Overall, using oversampling techniques ensures that the ML model is better equipped to handle imbalanced datasets, leading to enhanced predictive performance in soil movement prediction tasks.

Furthermore, the parameter analysis reveals that oversampling techniques add generalized information to the dataset, making it more consistent across different datasets. This reduced variability in the dataset allows ML models to learn these generalized patterns more effectively. As evident in the parameter analysis results, oversampling techniques lead to smaller stdev in parameter values across different models, indicating improved consistency and generalization. This further supports the notion that oversampling techniques help mitigate overfitting and enhance the overall performance of ML models in soil movement prediction tasks.

Some minor comments:

Figure 1 text is hard to read. Please increase the font size of figure 1

Thank you for your feedback. In the revised manuscript, we have replaced Figure 1 with a high-quality image and increased the font size for better readability. We appreciate your suggestion, and we believe that these improvements will enhance the clarity of the figure for readers.

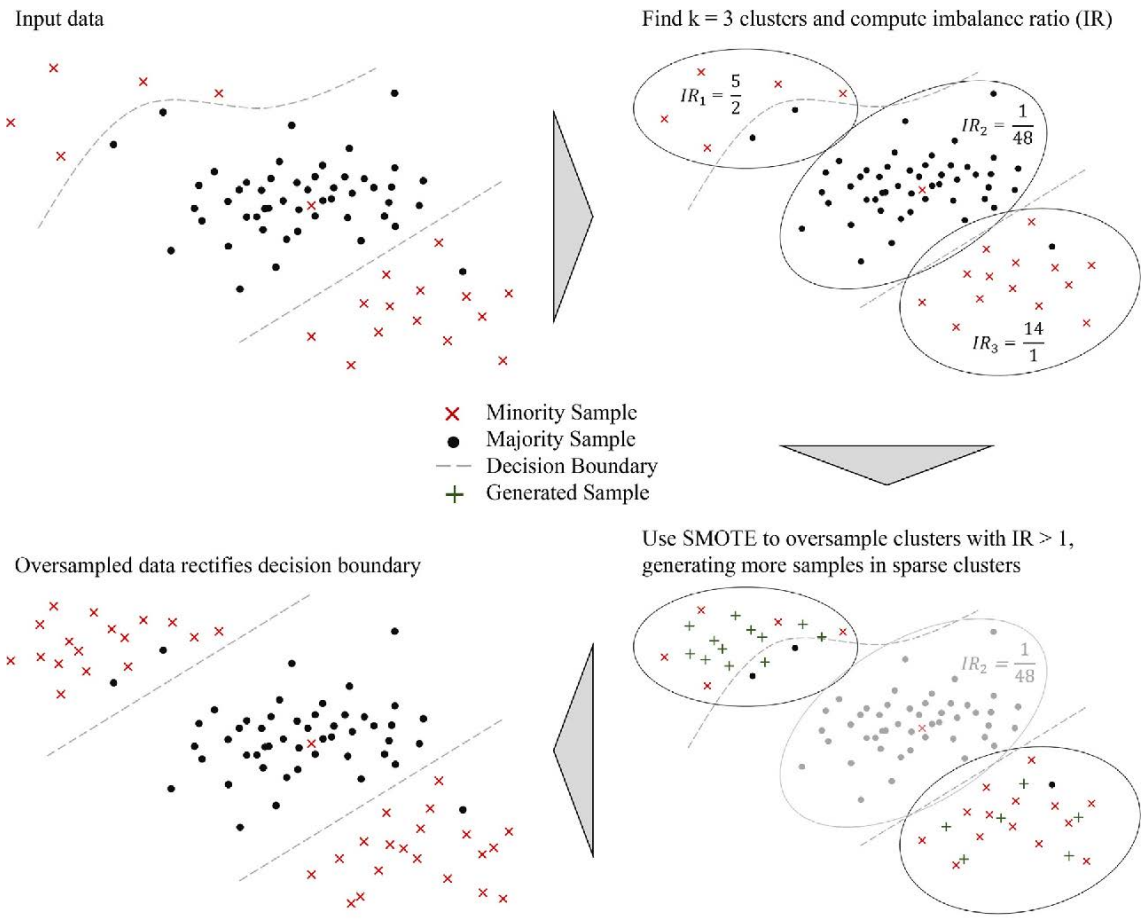

Figure 1. K-Means SMOTE effectively addresses within-class imbalance by oversampling safe areas (Douzas et al., 2018).