



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

2022년 8월
박사학위논문

머신러닝을 활용한 비탈면 안정성 평가 모델에 관한 연구

조선대학교 대학원

첨단에너지자원공학과

노 정 두

머신러닝을 활용한 비탈면 안정성 평가 모델에 관한 연구

Model for stability evaluation on a slope using
machine learning

2022년 8월 26일

조선대학교 대학원

첨단에너지자원공학과

노 정 두

머신러닝을 활용한 비탈면 안정성 평가 모델에 관한 연구

지도교수 강 성 승

이 논문을 공학 박사학위신청 논문으로 제출함

2022년 4월

조선대학교 대학원

첨단에너지자원공학과

노 정 두

노정두의 박사학위논문을 인준함

위원장	조선대학교	교수	장 일 식 (인)
위 원	충북대학교	교수	서 용 석 (인)
위 원	조선대학교	교수	김 대 현 (인)
위 원	한국지질자원연구원	연구원	나 태 유 (인)
위 원	조선대학교	교수	강 성 승 (인)

2022년 6월

조선대학교 대학원

차 례

차례	i
List of tables	iv
List of figures	vi
Abstract	x
1. 서론	1
1.1 연구배경 및 목적	1
1.2 비탈면 안정성에 관한 연구 동향	5
2. 분석 모형	10
2.1 머신러닝	10
2.1.1 머신러닝 알고리즘	10
2.1.2 분류 모델의 평가 지표	16
2.1.3 초매개변수 최적화	20
2.2 XGBoost	23
2.2.1 XGBoost 이론적 배경	23
2.2.2 XGBoost 학습 파라미터	33
2.3 LightGBM	34
2.3.1 LightGBM 이론적 배경	34
2.3.2 LightGBM 학습 파라미터	41
2.4 CatBoost	42
2.4.1 CatBoost 이론적 배경	42
2.4.2 CatBoost 학습 파라미터	49
3. 데이터 분석	50
3.1 비탈면 조사자료	50
3.2 범주형 데이터의 변수 구간화 및 인코딩	57

3.2.1 변수 구간화	57
3.2.2 변수 레이블링	67
3.3 수치형 데이터의 이상치 제거 및 스케일링	68
3.3.1 이상치 제거	68
3.3.2 스케일링	70
3.4 교차 분석	72
3.5 상관관계 분석	75
3.6 해석 단계 설정	77
4. 연구 결과	84
4.1 XGBoost 예측 모델	84
4.1.1 XGBoost 비탈면 안정성 예측 모델 초매개변수	84
4.1.2 XGBoost 비탈면 안정성 예측 모델 : Case-1	85
4.1.3 XGBoost 비탈면 안정성 예측 모델 : Case-2	89
4.1.4 XGBoost 비탈면 안정성 예측 모델 : Case-3	93
4.1.5 XGBoost 비탈면 안정성 예측 모델 결과	97
4.2 LightGBM 예측 모델	100
4.2.1 LightGBM 비탈면 안정성 예측 모델 초매개변수	100
4.2.2 LightGBM 비탈면 안정성 예측 모델 : Case-1	101
4.2.3 LightGBM 비탈면 안정성 예측 모델 : Case-2	105
4.2.4 LightGBM 비탈면 안정성 예측 모델 : Case-3	109
4.2.5 LightGBM 비탈면 안정성 예측 모델 결과	113
4.3 CatBoost 예측 모델	116
4.3.1 CatBoost 비탈면 안정성 예측 모델 초매개변수	116
4.3.2 CatBoost 비탈면 안정성 예측 모델 : Case-1	117
4.3.3 CatBoost 비탈면 안정성 예측 모델 : Case-2	121
4.3.4 CatBoost 비탈면 안정성 예측 모델 : Case-3	125
4.3.5 CatBoost 비탈면 안정성 예측 모델 결과	129
5. 토의	132
5.1 머신러닝 방법에 따른 비탈면 안정성 예측 모델 결과	132

5.2 비탈면 조사자료의 시각화를 통한 비탈면 관리 방안	148
6. 결론	151
참고문헌	154

List of tables

Table 2-1. Parameters mainly used in XGBoost	33
Table 2-2. Parameters mainly used in LightGBM	41
Table 2-3. Parameters mainly used in CatBoost	49
Table 3-1. Slope investigation data for 15 years in Korea	53
Table 3-2. The number of classes for each slope investigation contents	58
Table 3-3. Results for cross tabulation of categorical data	73
Table 3-4. Datasets of the LR-1 analysis model	78
Table 3-5. Datasets of the LR-2 analysis model	79
Table 3-6. Results for logistic regression according to the analysis steps	80
Table 3-7. Variable names and types according to each analysis step	83
Table 4-1. Results of hyperparameter tuning for XGBoost according to each analysis step	84
Table 4-2. Performance of Case-1 as a result of classification using XGBoost	86
Table 4-3. Performance of Case-2 as a result of classification using XGBoost	90
Table 4-4. Performance of Case-3 as a result of classification using XGBoost	94
Table 4-5. Results of hyperparameter tuning for LightGBM according to each analysis step	100
Table 4-6. Performance of Case-1 as a result of classification using LightGBM	102
Table 4-7. Performance of Case-2 as a result of classification using LightGBM	106
Table 4-8. Performance of Case-3 as a result of classification using LightGBM	110
Table 4-9. Results of hyperparameter tuning for CatBoost according to each analysis step	116
Table 4-10. Performance of Case-1 as a result of classification using CatBoost	118
Table 4-11. Performance of Case-2 as a result of classification using CatBoost	122
Table 4-12. Performance of Case-3 as a result of classification using CatBoost	126
Table 5-1. Result for prediction performance of slope stability model using Case-1 depending on machine learning methods	135
Table 5-2. Result for prediction performance of slope stability model using Case-2	

depending on machine learning methods 140

Table 5-3. Result for prediction performance of slope stability model using Case-3

depending on machine learning methods 145

List of figures

Figure 1-1. Tree structure of boosting based model (a) XGBoost, (b) LightGBM, (c) CatBoost	4
Figure 2-1. Example of artificial neural network	11
Figure 2-2. Example of deep neural network	12
Figure 2-3. Example of ensemble model	13
Figure 2-4. Example of boosting	14
Figure 2-5. Example of gradient boosting machine	15
Figure 2-6. Example of confusion matrix	17
Figure 2-7. Example of AUC-ROC curve	19
Figure 2-8. Schematic diagram of grid search and random search	21
Figure 2-9. Example of Bayesian optimization for learning rate parameter	22
Figure 2-10. Example of tree ensemble model (a) flow chart of tree based model, (b) first weak learner, (c) second weak learner, (d) prediction score in each leaf	24
Figure 2-11. Example of structure score calculation in XGBoost	28
Figure 2-12. Example of Exact Greedy and Approximate algorithm for split finding ..	29
Figure 2-13. Example of Sparsity-Aware split finding (a) initial dataset, (b) missing value goto right into in dataset, (c) missing value goto left in the dataset, (d) Tree structure with missing values for age and gender	31
Figure 2-14. Example of Histogram-based algorithm	35
Figure 2-15. Example of Gradient-based One Side Sampling (GOSS)	37
Figure 2-16. Example of Greedy Bundling (a) before bundling, (b) after bundling	39
Figure 2-17. Example of Merge Exclusive Feature	40
Figure 2-18. Ordered target statistics (a) before random permutation, (b) after random permutation	45
Figure 2-19. Ordered boosting according to random permutation (a) dataset before random permutation, (b) 1 st step of calculating gradient, (c) 2 nd step of calculating gradient, (d) 7 th step of calculating gradient	46

Figure 3-1. Top 10 values of lithology in the original dataset 57

Figure 3-2. Distribution of simplified categorical values on each categorical data 60

Figure 3-3. Distribution of each numerical data applied standardization 68

Figure 3-4. Distribution of standardized numerical data without outliers 69

Figure 3-5. Distribution of scaled numerical data 71

Figure 3-6. Correlation of variables in slope investigation data 76

Figure 3-7. AUC of LR1 as a result of logistic regression 80

Figure 3-8. AUC of LR2 as a result of logistic regression 81

Figure 3-9. Adjusting the imbalanced target value (a) before adjusting about data imbalanced (b) after adjusting about data imbalanced 82

Figure 4-1. Result for prediction performance of Case-1 using XGBoost 86

Figure 4-2. Result for AUC of Case-1 using XGBoost 87

Figure 4-3. Results for confusion matrix of Case-1 on (a) train data and (b) test data using XGBoost 87

Figure 4-4. Result for feature importance of Case-1 using XGBoost 88

Figure 4-5. Result for prediction performance of Case-2 using XGBoost 90

Figure 4-6. Result for AUC of Case-2 using XGBoost 91

Figure 4-7. Results for confusion matrix of Case-2 on (a) train data and (b) test data using XGBoost 91

Figure 4-8. Result for feature importance of Case-2 using XGBoost 92

Figure 4-9. Result for prediction performance of Case-3 using XGBoost 94

Figure 4-10. Result for AUC of Case-3 using XGBoost 95

Figure 4-11. Results for confusion matrix of Case-3 on (a) train data and (b) test data using XGBoost 95

Figure 4-12. Result for feature importance of Case-3 using XGBoost 96

Figure 4-13. Result for slope stability prediction model using XGBoost (a) precision, (b) recall (c) F1-Score, (d) accuracy 99

Figure 4-14. Result for prediction performance of Case-1 using LightGBM 102

Figure 4-15. Result for AUC of Case-1 using LightGBM 103

Figure 4-16. Results for confusion matrix of Case-1 on (a) train data and (b) test data using LightGBM 103

Figure 4-17. Result for feature importance of Case-1 using LightGBM 104

Figure 4-18. Result for prediction performance of Case-2 using LightGBM 106

Figure 4-19. Result for AUC of Case-2 using LightGBM 107

Figure 4-20. Results for confusion matrix of Case-2 on (a) train data and (b) test data using LightGBM 107

Figure 4-21. Result for feature importance of Case-2 using LightGBM 108

Figure 4-22. Result for prediction performance of Case-3 using LightGBM 110

Figure 4-23. Result for AUC of Case-3 using LightGBM 111

Figure 4-24. Results for confusion matrix of Case-3 on (a) train data and (b) test data using LightGBM 111

Figure 4-25. Result for feature importance of Case-3 using LightGBM 112

Figure 4-26. Result for slope stability prediction model using LightGBM (a) precision, (b) recall (c) F1-Score, (d) accuracy 115

Figure 4-27. Result for prediction performance of Case-1 using CatBoost 118

Figure 4-28. Result for AUC of Case-1 using CatBoost 119

Figure 4-29. Results for confusion matrix of Case-1 on (a) train data and (b) test data using CatBoost 119

Figure 4-30. Result for feature importance of Case-1 using CatBoost 120

Figure 4-31. Result for prediction performance of Case-2 using CatBoost 122

Figure 4-32. Result for AUC of Case-2 using CatBoost 123

Figure 4-33. Results for confusion matrix of Case-2 on (a) train data and (b) test data using CatBoost 123

Figure 4-34. Result for feature importance of Case-2 using CatBoost 124

Figure 4-35. Result for prediction performance of Case-3 using CatBoost 126

Figure 4-36. Result for AUC of Case-3 using CatBoost 127

Figure 4-37. Results for confusion matrix of Case-3 on (a) train data and (b) test data using CatBoost 127

Figure 4-38. Result for feature importance of Case-3 using CatBoost 128

Figure 4-39. Result for slope stability prediction model using CatBoost (a) precision, (b) recall (c) F1-Score, (d) accuracy 131

Figure 5-1. Result of prediction performance of slope stability model using Case-1

	depending on machine learning methods (a) precision, (b) recall, (c) F1-Score (d) accuracy	136
Figure 5-2.	Result of prediction performance of slope stability model using Case-2 depending on machine learning methods (a) precision, (b) recall, (c) F1-Score (d) accuracy	141
Figure 5-3.	Result of prediction performance of slope stability model using Case-3 depending on machine learning methods (a) precision, (b) recall, (c) F1-Score (d) accuracy	146
Figure 5-4.	Visualization of slope stability through national slope investigation data	149
Figure 5-5.	Visualization of slope length through national slope investigation data	150

Abstract

Model for stability evaluation on a slope using machine learning

By JEONGDU NOH

Adv. Prof. : Seong-Seung Kang, Ph.D,

Dept. of Advanced Energy Resources Engineering

Graduate School of Chosun University

In Korea, where more than 70% of the land is mountainous, the slopes are vulnerable due to topographical and meteorological influences. Traditional methods for determining slope stability include the theoretical method, statistical method, and numerical method. Recently, technologies such as artificial intelligence and big data have been introduced, and many studies using them have been conducted. However, it is considered that most studies are difficult to use generally because the number of data and variables is small. Therefore, this study suggests a prediction model for slope stability using more than 30,000 slope data and XGBoost, LightGBM, and CatBoost, which have been awarded and certified for excellence in various artificial intelligence competitions. For the purpose, first, input variables and output variable were selected through data pre-processing and statistical verification. Second, to understand the characteristics of slope investigation and to improve the performance of the prediction model for slope stability, analysis cases were set according to the data types of the input variables. Third, prediction models for slope stability using XGBoost, LightGBM, and CatBoost were compared. Finally, by summarizing the results, the prediction model for slope stability was cross-checked and proposed.

Slope data that were composed of 84 variables can be divided into basic survey, which are objective, and detailed surveys, which are subjective. Among these data, 31 input variables and 1 output variable were selected by performing outlier removal, binning, correlation analysis, statistical verification, and logistic regression. For carrying out analysis and comparison of feature importance and prediction model for slope stability, analysis steps were divided into Case-1, Case-2, and Case-3 using input data.

Case-1 was composed of 9 numerical data, and Case-2 consisted of 22 categorical data. Case-3 was formed of 31 numerical and categorical data. For each analysis step and machine learning model, training performance, AUC, prediction performance, and feature importance were estimated. The ratio of training data and test data was set to 7:3, and Bayesian optimization was used for hyper-parameter tuning. To prevent overfitting, k-fold cross-validation was used.

As a result of analyzing the prediction model for slope stability using XGBoost, AUC was found to be 0.668 for Case-1, 0.771 for Case-2, and 0.770 for Case-3. An analysis of the training and prediction performance for Case-1, Case-2, and Case-3 obtained good results. However, prediction imbalance occurred in precision and recall. Feature importance in Case-1 was similar to most of variables except for berm and valley. Feature importance in Case-2 was relatively higher in topography, bedrock shape, stability method 2, and weathering than other variables. Feature importance in Case-3 was higher in slope length, angle of upper slope, and soil depth than other variables.

As a result of analyzing the prediction model for slope stability using LightGBM, AUC was found to be 0.681 for Case-1, 0.766 for Case-2, and 0.783 for Case-3. An Analysis of the training and prediction performance for Case-1, Case-2, and Case-3 obtained the favorable results. Additionally, prediction imbalance did not occur. Feature importance in Case-1 was similar most of variables except for berm and valley. Feature importance in Case-2 was relatively higher bedrock shape and topography than other variables. Feature importance in Case-3 was higher in soil depth, angle of upper slope, slope angle, slope length, slope height, and distance from the road than other variables.

As a result of analyzing the prediction model for slope stability using CatBoost, AUC was found to be 0.672 for Case-1, 0.777 for Case-2, and 0.794 for Case-3. An analysis of the training and prediction performance for Case-1, Case-2, and Case-3 obtained the highest results. Feature importance in Case-1 was higher in soil depth, angle of upper slope, and slope height than other variables. Feature importance in Case-2 was higher in soil depth, the direction of discontinuities, the shape of slope side, and type of discontinuities than other variables.

Although XGBoost showed good performance in the training and prediction, there

was an imbalance in precision and recall. Feature importance was found to be relatively higher in the numerical data than in the categorical data using XGBoost. LightGBM showed good performance in the training and prediction. However, prediction imbalanced in precision and recall did not occur. Feature importance was found to be relatively higher in the numerical data than in categorical data using LightGBM. CatBoost showed the best performance in the training and prediction, but overfitting with train data occurred. Feature importance was found to be relatively higher in the categorical data than in the numerical data.

Putting together, based on the prediction performance of the model, it was found that categorical data played an important role in evaluating slope stability based on prediction performance than numerical data. Still, it was also likely to cause overfitting in training performance. Therefore, it is considered that XGBoost, LightGBM, and CatBoost can be used all as prediction models for slope stability. but among them, LightGBM is deemed to be able to predict slope stability most stably, and CatBoost is considered to predict slope stability most accurately.

1. 서론

1.1 연구배경 및 목적

인류는 생활의 편의성을 위해 끊임없이 기술을 발전시키고 있다. 전자기기를 통해 생활 대부분을 통제할 수 있으며, 기계 스스로 교육하여 사용자가 관여하지 않아도 임무를 수행한다. 이러한 기술 발전은 인류, 자연환경, 동·식물 등에 부정적인 영향을 미치기도 한다. 자연환경의 부정적인 요인으로는 환경오염, 지구온난화, 가뭄, 그리고 폭우 등을 예로 들 수 있다. 특히, 폭우와 집중호우에 의한 산사태, 홍수, 지반침하하는 지속적으로 증가하는 추세이다. 2021년 여름 미국을 포함한 북미는 폭염, 서유럽과 서호주, 그리고 동북아시아는 집중호우와 태풍에 의한 인명 및 재산피해가 발생하였다. 전국토의 약 70%가 산지인 우리나라는 인구 증가, 도시화, 도로 확장, 국토 효율화 등의 이유로 대규모 산지 개발의 빈도가 점점 더 증가하고 있기 때문에, 이것에 의한 자연환경에 취약하게 노출된 상태이다. 2021년 산림청 통계에 따르면 산사태 발생 빈도수는 해마다 증가하고 있으며, 특히 여름철 집중되는 태풍 및 집중호우의 영향으로 산사태 발생 규모가 시기별 및 지역별로 편중되어 나타나는 경향이 있음을 보고하였다(Korea Forest Service, 2021).

산사태 발생을 판단하고 예측하는데 사용되는 전통적인 방법으로는 이론적 해석 및 통계적 해석을 들 수 있다. 산사태 관련 연구에 가장 많이 사용되고 있는 방법으로는 산사태를 유발시키는 성향을 정의하고 대상 지역을 확률적으로 표현하는 산사태 취약성 분석을 들 수 있다(Guzzetti et al., 2006; Guzzetti et al., 2007). 산사태 취약성은 각 분석 요소를 등급화하여 산사태 취약성 지도를 작성해 관리한다. 산사태 취약성 지도를 제작하는 방법은 크게 정성적 및 정량적 해석기법으로 구분할 수 있다. 초기 산사태 취약성 분석의 불확실성을 고려하기 위해 조건부 확률, 증거비중 (Weight of Evidence, WOE), 빈도비 (Frequency Ratio, FR), 로지스틱 회귀분석 (Logistic Regression, LR) 등이 사용되었고, 2010년에 들어서 인공신경망 (Artificial Neural Networks, ANN), 딥러닝 (Deep Learning, DN) 등의 방법에 대한 다양한 연구가 많이 진행되었다. Chung and Fabbri(1999)는 조건부 확률과 확신도 (Certainty Factor, CF) 등을 이용하였고, Suzen and Doyuran(2004)는 WOE를 이용하였다. Pham et al.(2016)은 구글 이미지로 분석한 930개 산사태 발생지역

에 대하여 Support Vector Machine (SVM), LR, Fisher's Linear Discriminant Analysis (FLDA), Bayesian Network (BN), Naive Bayes (NB)를 이용해 산사태 취약성을 분석하였고, Wang et al.(2016)은 LR, FR, Decision Tree (DT), WOE, ANN을 이용해 산사태 예측 모델을 제안하였다. Chen et al.(2017)은 DEM 자료와 Landsat-8의 자료를 토대로 Random Forest (RF)을 통해 산사태 취약성 모델을 제안하였다. Xiao et al.(2020)은 하이브리드 모델로서 RF, FR, CF, Index of Entropy (IOP)를 각각 결합한 RF-FR, RF-CF, RF-IOE, IOE-CF, CF-FR을 이용해 산사태 취약성 모델을 제안하였다.

최근에는 더 진보된 기술의 발전으로 빅데이터, 머신러닝 (machine learning) 및 딥러닝 (deep learning)이 기존의 수학적, 공학적 한계점을 극복할 수 있는 방법 중 하나로 인식되었다. 예측 모델의 정확도를 향상시키기 위한 노력은 꾸준히 진행되고 있으며, 대표적인 예측 모델로 부스팅 기반 모델인 XGBoost, LightGBM, CatBoost 등이 있다(Chen and Guestrin, 2016; Ke et al., 2017, Prokhorenkova et al., 2017). XGBoost (eXtreme Gradient Boosting)는 GBM (Gradient Boosting Machine)을 기반하는 Boosting의 대표적인 기법으로 2015년 Kaggle의 총 29개 대회에서 17개가 XGBoost를 사용해 우승하였다. XGBoost는 기존 Boosting 기반의 트리 모델 (tree model)과 달리 효율적인 계산을 위한 Weighted quantile sketch, 병렬 학습을 위한 Novel sparsity aware 알고리즘과 트리를 학습하기 위한 효율적 캐시 인식 블록 구조를 제안하였다(Chen and Guestrin, 2016). LightGBM (Light Gradient Boosting Machine)은 Microsoft에서 개발한 모델로 잎 분할 (leaf wise) 방식을 사용하여 기존의 트리 구조 모델들의 방식과 다르며, GOSS (Gradient-based One-Side Sampling)와 EFB (Exclusive Feature Bundling)를 통해 데이터 차원을 줄여 보다 정확한 모델을 만들고 학습 속도를 높이는 방식을 사용하고 있다(Ke et al., 2017). CatBoost (Category Boost)는 Yandex에서 개발한 방법으로 범주형 데이터 처리와 훈련 과정에서 발생하는 예측 변화 (prediction shift)를 해결하기 위해 순차적 부스팅 (ordered Boosting)을 사용하였다(Prokhorenkova et al., 2017). 이 세 가지 트리 기반 앙상블 모델의 구조는 Figure 1-1과 같고, 다양하고 많은 양의 데이터가 요구된다. 이 3개의 모델은 여러 인공지능 대회에서 높은 성능으로 여러 차례 우승하였고, 다양한 연구 분야에서 그 사용 빈도가 증가하고 있다. 대표적인 연구로 Yoon(2020b)은 심혈관질환을 분석하기 위해 LightGBM과 심전도 자료를 이용해 정상 심박 분류에 높은 해석력을 나타냈다. Kim(2020)은

주택보유기간에 영향을 미치는 결정요인을 추정하기 위해 Random Forest, XGBoost, LightGBM 등의 머신러닝 모델들을 이용하여 비교분석을 수행하였다. Yoon(2020a)은 XGBoost, Random Forest를 이용한 공기업 재무 건전성을 예측하였다. An(2021)은 기업혁신과 경영 성과를 예측하기 위해 XGBoost, LightGBM, CatBoost를 이용하여 비교 분석하였다. Lee(2021)는 머신러닝 모듈인 XGBoost, LightGBM, Random Forest와 입력변수 39개를 이용한 토지 피복 분류 자동화 모델의 성능을 비교 분석하였다. Kim(2021)은 인터넷상에서 유포되는 악성코드의 변종들을 분류 및 탐지하기 위해 LightGBM을 이용하여 악성코드 탐색 및 분류하는 시스템을 제안했다. Cheon et al.(2021)은 교통 혼잡 예측 모델을 제안하기 위해 여러 개의 머신러닝과 딥러닝을 이용하였고, 그 중 CatBoost가 가장 높은 성능을 나타냈다. 지금까지 머신러닝 및 딥러닝을 이용한 선행연구들은 여러 학습 지표와 평가 자료를 통해 높은 정확도와 예측률을 나타냈지만, 제한된 데이터 수와 입력변수로 인해 낮은 범용 가능성과 신뢰성을 보여주기도 한다.

앞서 설명한 비탈면 안정성 예측에 관한 대부분의 연구는 위성 및 항공사진, DEM 등의 비정형 데이터와 10개 이하의 현장 조사자료로부터 얻은 변수들이 이용되었다. 또한 산사태 자료는 대부분 약 300개가 이용되었다. 따라서 선행연구에서 제안된 모델들은 지질학적 및 지형학적 특성을 충분히 반영되었다고 보기에 어려움이 있어 보이므로, 보편적으로 사용하는 것에도 무리가 있을 수 있다. 이러한 점을 고려하여 비탈면의 다양한 조건과 현황에 적용 가능한 비탈면 안정성 예측 모델을 개발함으로써 비탈면을 효율적으로 관리할 수 있는 방안을 제안하고자 한다. 이러한 목적을 달성하기 위하여, 첫째, 2006년부터 2020년에 수집된 국내의 약 30,000여 개의 비탈면 조사자료를 각 특성에 대해 이상치를 제거와 인코딩 등의 데이터 전처리 과정을 수행하였다. 둘째, 비탈면 자료는 크게 사면의 객관적 정보인 일반현황과 전문가의 주관적 판단인 정밀 조사자료로 구성되어 있다. 비탈면 안정성에 영향을 미치는 특성을 파악하고 통계적 분석을 실시하였고, 비탈면 안정성 평가 모델의 성능을 향상시키기 위해 사면 조사자료의 구성에 따른 해석 단계를 설정하였다. 셋째, 머신러닝의 앙상블 대표 모델인 XGBoost, LightGBM, CatBoost를 이용해 비탈면 안정성 예측 모델을 비교하고 분석하였다. 마지막으로 이들 결과를 종합하여 비탈면 안정성 예측 모델을 교차 검증하였다.

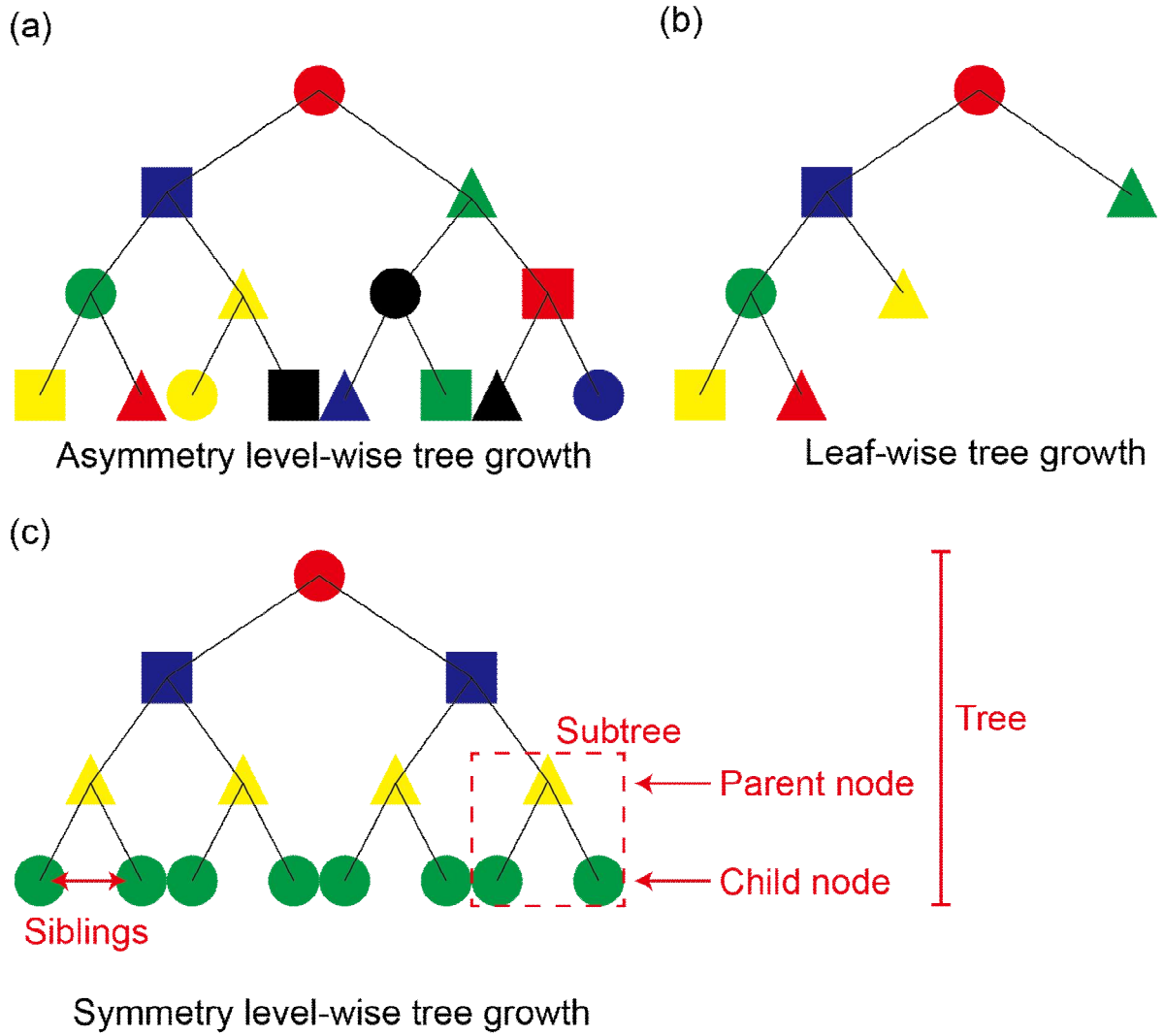


Figure 1-1. Tree structure of boosting based model (a) XGBoost, (b) LightGBM, (c) CatBoost.

1.2 비탈면 안정성에 관한 연구 동향

산사태는 산지의 급사면을 구성하는 물질이 하부로 급격히 이동하는 현상을 말한다. 급경사의 사면이 자중으로 붕괴되는 사면붕괴, 전단 응력에 의해 약한 면을 따라 파괴가 일어나는 활동파괴, 폭우 시 급사면의 토사가 느슨해져 물과 함께 흘러내리는 토석류, 고운 입자 크기의 토양과 암석이 붕괴하는 이석류, 중력에 의해 암석이 탈락하는 낙석 등을 산사태로 나눌 수 있다(USGS, 2008; Wikipedia, 2022d). 산사태는 내부적 요인과 외부적 요인에 의해 발생한다. 내부적 요인으로는 지형적 및 지질학적 조건 등이며, 외부적 요인으로는 기상, 지진, 사회적 조건 등이 이에 해당한다. 산사태는 무분별한 개발로 지반이 약해지는 경우가 아니면 주로 외부적 요인에 의해 갑작스럽게 발생한다(Wiezorek, 1994; Skilodimou et al., 2018; Yang et al., 2020; Getachew and Meten, 2021). 산사태가 발생하면 대규모의 인명 및 재산피해가 발생하기 때문에 사전에 예측해 피해를 최소화하는 것이 중요하다.

기존의 산사태 연구는 주로 이론에 근거한 접근법, 통계적 방법, 수치해석 방법 등이 사용되었다. 대표적으로 Caine(1980)은 토석류를 유발하는 임계강우개념(Critical rainfall threshold concept)을 제안하였다. Lee et al.(2002)은 지형, 토양, 임상, 토지 이용, 지질구조 등을 점수화해 광역적 산사태 취약성을 분석하였다. Kim et al.(2005)은 GIS (Geographic Information System)와 RS (Remote Sensing technique)를 이용해 산사태 취약성을 분석하였다. Kim et al.(2008)은 비탈면의 효율적 관리를 위해 약 2,500여 개의 비탈면에 대해 정밀조사를 실시하였고, 지역별 위험 사면 분포도를 제시하였다. Xu et al.(2011)은 산사태 취약성에 대하여 민감성(sensitivity), 노출(exposure), 적응성(adaptation capacity)으로 구분하여 평가하였다. Ko et al.(2014)은 LiDAR를 이용하여 도출해낸 지형 인자들에 대한 단위면적당 산사태 발생 빈도를 분석하였다. Oh and Park(2014)은 산사태 발생을 예측하기 위해 지역적, 지질학적 특성과 강우강도-지속시간(Intensive-Duration method, I-D method)을 고려하여 강우 기준을 결정하였다. Lee et al.(2015)은 10년간 발생한 산사태의 지형 특성을 이용하여 산사태 예측 방법을 비교 분석하였다. Cha et al.(2015)은 지형 특성과 산사태 위험도의 상관관계를 분석하고자 벡터 분산과 평면도 기법을 사용하였다. Yoon et al.(2015)은 국내 풍화도에 대한 실내 강우 침투실험 결과와 수치해석 프로그램을 이용해 불포화 사면의 안정성을 분석하였다. Hong

et al.(2016)은 선행강우의 영향을 고려한 국내 산사태 유발강우기준(Intensity-Duration, ID curve)을 제안하였다. Seo et al.(2017)은 극한 강우에 의한 비탈면의 거동을 파악하기 위해 국내 산사태 발생 빈도가 높은 화강암 풍화토, 편마암 풍화토로 이루어진 산사태 모형을 제작하여 강우 침투에 의한 비탈면의 거동, 모관흡수력과 체적함수비의 변화를 분석하였다. Kim et al.(2017)은 산지 내 임도 개설을 위해 극한 강우의 침투 특성을 고려하여 사면안정해석을 수행하였다. Kwak et al.(2018)은 강우변화에 따른 강우의 토층 내 침투깊이를 고려한 산사태 위험지수를 개발하였다.

최근에는 빅데이터와 인공지능을 활용한 비탈면 분야의 연구가 활발하게 진행되고 있다. 이것은 크게 3가지로 분류할 수 있는데, 첫째는 모니터링 자료를 활용해 비탈면의 거동을 예측하는 것이다. 둘째는 강우강도에 의한 비탈면의 안정성 분석 또는 강수량과 강우발생빈도를 예측하여 산사태 발생을 판단하는 것이다. 마지막으로 항공사진, 위성사진 등의 비정형 데이터와 비탈면 조사자료를 이용한 비탈면 안정성을 예측하는 것이다. 대표적인 연구로 Zhou et al.(2018)은 hybrid ML (Machine Learning)을 비탈면의 변위를 예측하여 산사태 발생 가능성에 대해 조사하였고, Yang et al.(2019)은 Long Short Term Memory (LSTM)를 이용하여 비탈면의 변위를 예측 산사태 발생 여부를 판단하였다. Gao et al.(2019)은 Gray System Method (GSM)과 Evolutionary Neural Network (ENN)을 결합한 모델로 비탈면의 변위를 예측한 자료를 이용해 산사태 발생 여부를 분석하였다. Krkac et al.(2020)은 조기 경보 시스템과 비탈면의 움직임 예측하기 위해 비탈면 모니터링 자료와 지하수위, 강수량 자료를 RF와 multiple linear regression을 사용해 비탈면 안정성 예측 모델을 제안하였다. Hong et al.(2019)는 급경사지 붕괴 이력과 유발강우량 분석을 통하여 일 누적 강우량과 최대 시간 강우량이 급경사지 붕괴와 관련이 높다고 분석하였다. Kim et al.(2020)은 613개 산사태 데이터를 통해 I-D threshold를 분석한 결과, 한반도 남쪽 지역은 강우에 의한 산사태 발생에 취약한 것으로 나타났다. Gomez and Kavzoglu(2005)는 DEM, remote sensing imagery, 현장 조사자료를 ANN으로 분석해 산사태 예측 모델을 제안하였다. Melchiorre et al.(2008)은 ANN과 Cluster analysis (CAF)를 사용하여 산사태 취약성 분석 모델을 제안하였다. Saito et al.(2009)는 8개 비탈면의 변수와 DT를 이용해 산사태 취약성을 분석하였다. Pradhan et al.(2010)은 비탈면의 15개 인자를 ANN으로 분석해 산사태 취약성 모델을 제안하였다. Yeon(2011)은 산사태 취약성 분석을 위해 로

지스틱 회귀분석 기법과 비탈면의 지형적 형태, 지질학적 요소 등을 고려하여 예측 성능을 분석하였다. Lee and Kim(2012)는 현장 조사자료와 항공사진, DEM 자료를 분석하여 취득한 자료를 바탕으로 로지스틱 회귀분석을 실시하였다. Bui et al.(2016)은 Bagging, Adaboost, MultiBoost를 이용해 산사태 취약성 모델을 제안했다. Su et al.(2017)은 SVM, LR, ANN을 이용해 산사태 취약성 모델을 제안했으며, 암종, 도로부터의 거리, 사면각도, 높이, 사면 이용 종류가 변수 중요도가 높게 나타났다. Kim et al.(2017)은 17개의 비탈면 특성으로 구성된 2,100개의 산사태 자료와 RF, Boosted tree model을 사용하여 산사태 취약성 모델을 제안하였고 Topographic Wetness Index (TWI), Slope gradient, Slope aspect 순으로 변수 중요도가 높았다. Liu and Wu(2018)은 Auto Encoder, ANN, SVM을 이용해 지질재해 인식 모델을 제안하였다. Huang et al.(2019)는 약 300개 산사태 자료의 27개 환경 영향 요소를 대상으로 Fully Connected Sparse Auto Encoder (FC-SAE), SVM, Back Propagation Neural Network (BPNN)을 이용하여 분석해 산사태 취약성 예측 모델을 제안하였다. Di et al.(2019)는 약 3,800개 토석류 자료를 LR, K-Nearest Neighbor (KNN), SVM, ANN, GBM을 이용해 산사태 취약성 모델을 제안하였고, 변수 중요도는 강수량, 고도, 건조지수 순으로 높게 나타났다. Dao et al.(2020)은 217개 산사태 자료에 대한 9개 산사태 발생 요인의 데이터 셋을 확보하고 Quadratic Discriminant Analysis (QDA), FLDA, Multilayer Perceptron Neural Network (MLP), Deep Learning (DL)로 산사태 취약성에 대해 분석하였다. Sahin(2020)은 Gradient Boosting 알고리즘 계열의 GBM, XGBoost, LightGBM, CatBoost을 이용해 산사태 취약성을 비교 분석하였다. Nhu et al.(2020)은 66개 얇은 산사태 자료를 SVM, Bayesian Logistic Regression (BLR), Alternating Decision Tree (ADTree)를 이용해 산사태 취약성 모델을 제안했다. Wang et al.(2021)은 지형자료, 지질자료, 강우 자료를 3가지 데이터셋으로 구분하여 LR, SVM, RF, CNN, Boosting model로 산사태 취약성 모델을 제시하였다. Nhu et al.(2020)은 152개 산사태 자료의 17개 비탈면 변수와 Adaboost, ADTree, 그리고 Adaboost-ADTree 모델을 이용해 산사태 예측 모델을 제안하였다. Pradhan et al.(2018)은 260개의 산사태 자료와 그 중 35개 자료의 산사태 발생시간, 그리고 해당 시간의 강우 강도 및 강우 기간을 데이터셋에 구성하여 분석하였다. 산사태 취약성 분석을 위해 Topographic factors, Hydrology factors, Soil factors, Forest factors, Geological factors와 ANN을 사용하여 산사태 발생을 분석하였고, 이를 국

립재난안전연구원(National Disaster Management Institute, NDMI)에서 제시한 강우의 위험수준 및 대응방안과 결합해 산사태 취약성 모델을 제안하였다. 이상의 연구들은 앞서 설명했듯이 비탈면 안정성 예측 모델을 비정형 데이터와 10개 이하의 현장 조사자료를 사용하였다. 대규모 단위로 비탈면 안정성을 예측 및 관리하거나 보편적으로 사용하는데 무리가 있는 것으로 판단된다.

2019년 이후 본 연구와 유사한 데이터를 이용한 연구가 Nam et al.(2020), Woo et al.(2020a), Woo et al.(2020b), 그리고 Lee et al.(2021)에 의해 수행되었다. Nam et al.(2020)은 2007년부터 2020년까지 조사된 30,165개 비탈면 현장 조사자료와 DNN, Bagging, Distribution Random Forest (DRF), extremely Randomized Trees (XRT), GBM, XGBoost를 이용하여 산사태 예측 모델을 비교 분석하였다. 이를 위해 지형학적 변수 8개와 지질학적 변수 6개 등을 입력데이터로 활용하였다. 출력데이터는 산사태 발생지역과 미발생지역으로 구분되는 이진변수를 사용하였고, 훈련 데이터와 시험데이터 비율은 7:3으로 설정하였다. 그 결과 XRT가 예측 성능이 가장 좋았고, 다음으로 GBM, DRF, XGBoost, DNN, LR순으로 나타났다. Woo et al.(2020a)은 도로비탈면관리시스템을 이용하여 해당 비탈면을 LR을 이용해 ‘조치’에 대한 예측 모델을 구축하고 검증하였다. 입력데이터는 ‘행정구역’, ‘왕복/편도’, ‘암반형태’, ‘불연속면 종류’ 등 총 29개로 구성되었다. 출력데이터는 전문가의 주관적 판단 자료 중 비탈면의 ‘응급’과 ‘미응급’을 나타내는 ‘조치’를 사용하였다. 훈련 데이터와 시험데이터의 비율은 9:1로 설정하여 이진분류 로지스틱 회귀분석을 실시하였고, 예측 성능은 정확도 95.29%로 나타났다. Woo et al.(2020b)은 강원도와 경상도 지역의 도로 비탈면 조사보고서와 정규 크리깅을 이용해 분석한 일 단위 기상 데이터를 Adaboost를 통해 비탈면 안정성 예측 모델을 제안하였다. 비탈면 현장 조사자료 중 해빙기 조사가 아닌 수시 조사 711건을 대상으로 한정하였고, 비탈면의 기본적 제원 정보와 기타 붕괴 발생과 관련한 인자를 사용하였다. 기상자료는 기상청에서 제공하는 일 단위 종관기상관측 데이터를 이용하였다. 학습데이터와 시험데이터 비율은 9:1로 설정하였고, 예측 모델의 정확도는 94.49%로 나타났다. Lee et al.(2021)은 2006년부터 2019년까지 조사된 도로비탈면관리시스템의 비탈면 데이터와 LR, DNN을 활용해 변수의 결측치를 예측하여 해당 결측값을 보완할 수 있는 예측 모델을 비교분석 하였다. 이를 위한 입력데이터로 Woo et al.(2022a)와 동일한 29개의 입력데이터를 사용하였고, 출력데이터는 3개 이상의 클래스를 갖는 변수들을 사용하였으며, 훈련 및 시험데이터의 비율은 19:1로 설정하였다. 다항 로지스

틱 회귀모델의 예측 성능 결과 대부분 LR 보다 DNN이 더 높은 정확도를 나타냈다. 하지만 이러한 연구들은 모델 학습에 비탈면 위치 정보와 도로 현황 등 비탈면 특성과 관련 없는 데이터를 사용하거나, 붕괴 발생과 관련한 인자를 사용하였다. 또한 예측 성능 평가를 1가지 또는 2가지 지표를 이용하였으며, 모델 학습시 검증 데이터 비율이 현저히 낮다는 단점이 있다.

2. 분석 모형

2.1 머신러닝

2.1.1 머신러닝 알고리즘

머신러닝은 데이터를 학습시켜 모델을 만드는 기법으로 명시적 프로그래밍 없이 컴퓨터 스스로 학습하는 능력을 부여하는 것이다. 명시적 프로그래밍은 개발자가 규칙을 정해놓고 그것에 맞게 반응하도록 하는 프로그래밍 기법이다. 명시적 프로그래밍은 간단하게 수행할 수 있지만, 데이터가 많아지고 복잡해지면 문제를 해결하기 어렵다는 단점이 있다. 따라서 이러한 명시적 프로그래밍의 한계를 극복하기 위해 적용되는 방법이 컴퓨터가 스스로 학습하도록 하는 머신러닝이다(Yoon et al., 2018; Oh, 2020).

머신러닝 기법은 학습 형태에 따라 지도 학습, 비지도 학습 등으로 분류할 수 있다. 그중 지도 학습은 학습데이터가 입력데이터와 출력데이터의 쌍으로 구성되어야 하는 기법으로, 대표적인 지도 학습의 기법은 인공신경망, 심층신경망, 앙상블 모델 등을 들 수 있다. 또한 지도 학습은 모델의 쓰임새에 따라 분류와 회귀로 나눌 수 있다. 분류는 학습데이터의 출력데이터로 범주를 학습시켜 입력데이터가 어떠한 범주에 속하는지 찾아내는 모델이다. 회귀는 입력데이터와 예측하려고 하는 값인 출력데이터로 모델을 학습시켜 새로운 입력데이터에 대한 출력값을 예측하는 모델이다. 비지도 학습은 입력데이터만 있는 훈련데이터를 이용하여 모델이 스스로 규칙성을 찾는 학습 방법이다. 준지도 학습은 훈련데이터의 일부만 정답이 있는 경우에 해당하는 학습 방법이다(Kim, 2016; Oh, 2020).

(1) Artificial Neural Network(ANN)

인공신경망(Artificial Neural Network, ANN)은 인간의 신경계와 비슷한 성능과 특성을 갖는 정보처리 시스템의 하나이다. 인공신경망은 인간의 뇌가 대량의 데이터를 효율적이고 병렬적으로 학습하고 처리할 수 있다는 사실을 기반으로 인간의 신

경세포를 컴퓨터를 통해 모델링하여 구현하는 방식이다(Fausett, 1994; Oh, 2020).

인공신경망은 Figure 2-1과 같이 입력층, 은닉층, 출력층으로 구성되어 있다. 인공신경망에서 각각의 층에 존재하는 인공뉴런을 노드라고 칭한다. 노드를 연결하는 선은 가중치를 가지고 있다. 입력층은 외부에서 입력받은 신호를 은닉층으로 전달하는 역할을 수행한다. 입력층과 출력층 사이에는 은닉층이 존재하고 있으며, 은닉층은 입력층으로부터 전달받은 데이터를 처리하여 출력값을 결정하고, 이를 출력층으로 전달하는 역할을 수행한다. 즉, 인공신경망은 외부에서 받은 입력데이터에 대해 가중치를 곱한 합을 내부 활성화함수를 거쳐 출력하여 은닉층 노드와 출력층 노드의 출력값을 출력한다. 출력에 사용되는 수식은 식 (2.1)과 같다. w 는 가중치, b 는 편향(bias)이다. 모델의 훈련은 순전파로 진행되며 모델 학습을 위한 가중치 업데이트는 역전파를 이용한다(Olivares, 2012; Kang, 2018; Oh, 2020).

$$y = h(b + w_1x_1 + w_2x_2) \tag{2.1}$$

$$y = h(a)$$

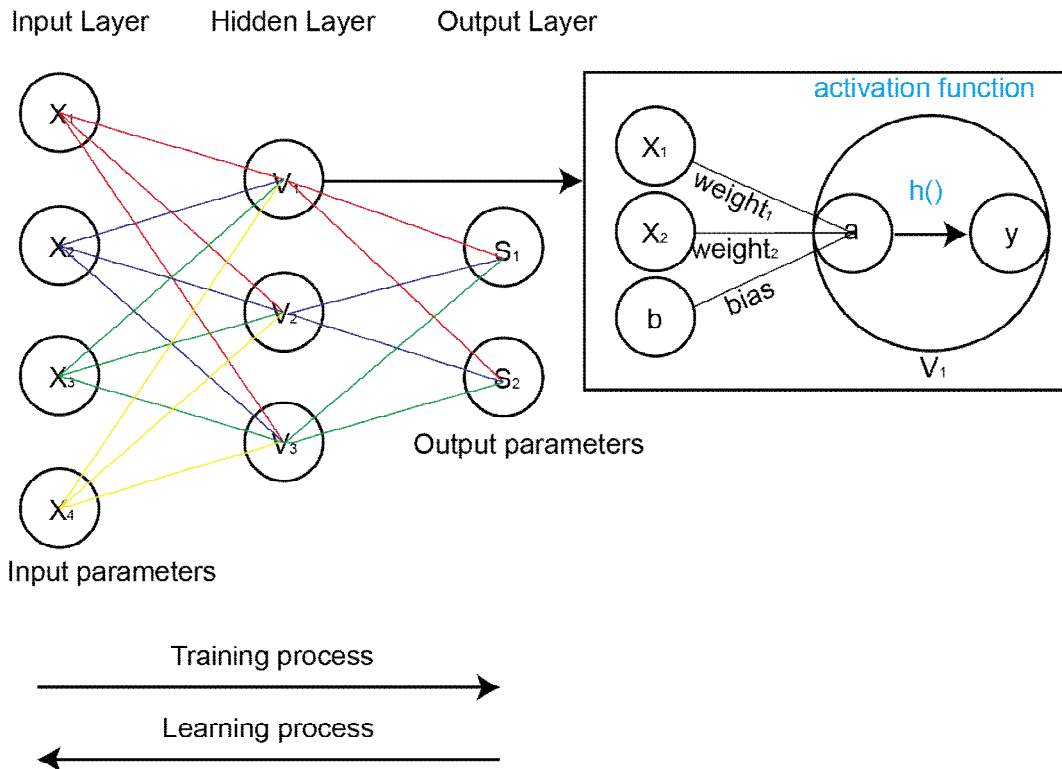


Figure 2-1. Example of artificial neural network.

(2) Deep Neural Network(DNN)

단순한 데이터가 아닌 수 만개의 데이터를 학습시키기 위해서는 하나 또는 두 개의 은닉층이 있는 인공신경망으로는 학습이 불가능하다. 따라서 은닉층의 개수를 수십 개 이상으로 늘린 복잡한 구조를 가진 신경망을 사용해야 한다. 이와 같이 깊고 복잡한 신경망을 이용하는 머신러닝 기법을 딥러닝이라 한다. 심층신경망(Deep Neural Network, DNN)은 딥러닝 기법 중 하나로 앞에서 설명한 인공신경망에서 은닉층 2개 이상으로 늘려 깊어진 구조를 말한다(Yoon et al., 2018). Figure 2-2는 심층신경망의 구조를 나타낸 것이다.

인공신경망을 여러 층으로 겹치는 것은 인공신경망의 선형성에 의해 하나의 층을 사용하는 것과 차이가 없다. 여러 개의 층을 사용할 때, 이를 의미 있게 하기 위해서는 각 층을 지날 때 마다 비선형 활성화 함수를 적용해야 한다. 비선형 활성화 함수로는 시그모이드 함수와 ReLU 함수 등이 주로 사용된다. 식 (2.2)은 l 번째 층에서 이전 층의 출력 y^{l-1} 이 입력데이터로 주어질 때 출력데이터 y^l 을 계산하는 과정을 나타낸 것이다. 여기서 σ 는 활성화 함수를 의미한다(Kim, 2016; Oh, 2020).

$$y^l = \sigma(w^l y^{l-1} + b^l) \quad (2.2)$$

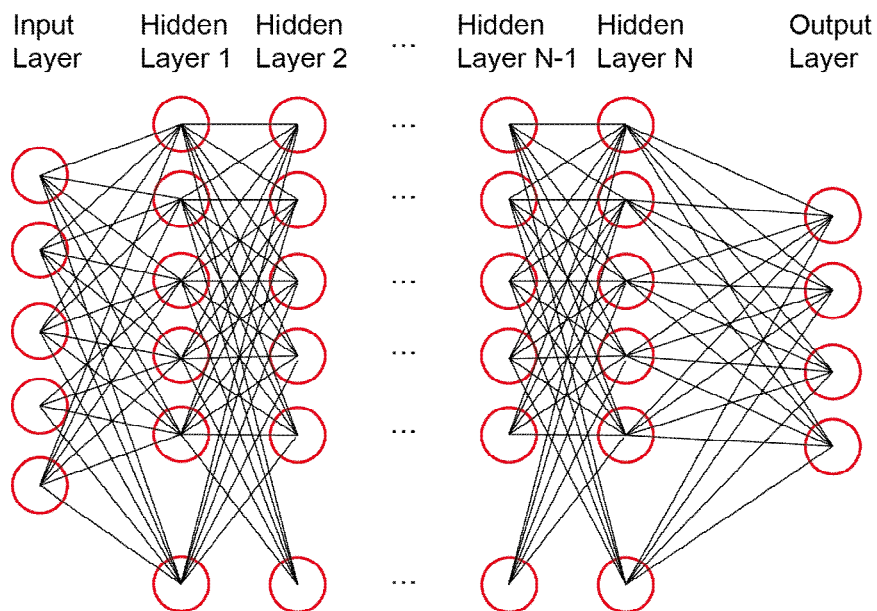


Figure 2-2. Example of deep neural network.

(3) Ensemble model

앙상블이란 여러 분류기로부터 얻은 복수의 의견을 적절한 방법으로 결합하여 좀 더 적합한 의사결정을 하는 접근방법이다. 앙상블을 사용하는 가장 중요한 이유는 가장 뛰어난 예측기 하나 보다 여러 개의 예측기로 이루어진 앙상블 모델의 성능이 더 우월하기 때문이다. 앙상블 모델(ensemble model)의 모식도는 Figure 2-3 과 같고, 분류와 회귀에 모두 적용할 수 있다. 여러 의견은 결합할 때 의견의 다양성이 매우 중요하다. 다양한 개별분류기를 생성하는 방법에는 여러 가지가 있는데, 주로 훈련(training) 집단을 재표본 방법(resampling method)와 특성(feature)의 공간 일부를 선택하는 부분 공간 방법(subspace method)이 사용된다. 재표본 방법은 훈련 집합에서 $p\%$ 를 임의로 선택하는 일을 여러 번 반복하여 서로 다른 훈련 집합을 여러 개 만들고, 이 훈련 집합들로 여러 개의 개별분류기를 만드는 방법이다. 부분 공간 방법은 d 차원의 원래 특징 공간에서 d 보다 작은 수의 특징을 임의로 선택하는 일을 반복하여 여러 개의 부분 공간을 구성한 후, 이 부분 공간에서 학습을 수행하여 개별분류기를 여러 개 만든다. 대표적인 앙상블 방법은 부스팅이 있다 (Wikipedia, 2022b).

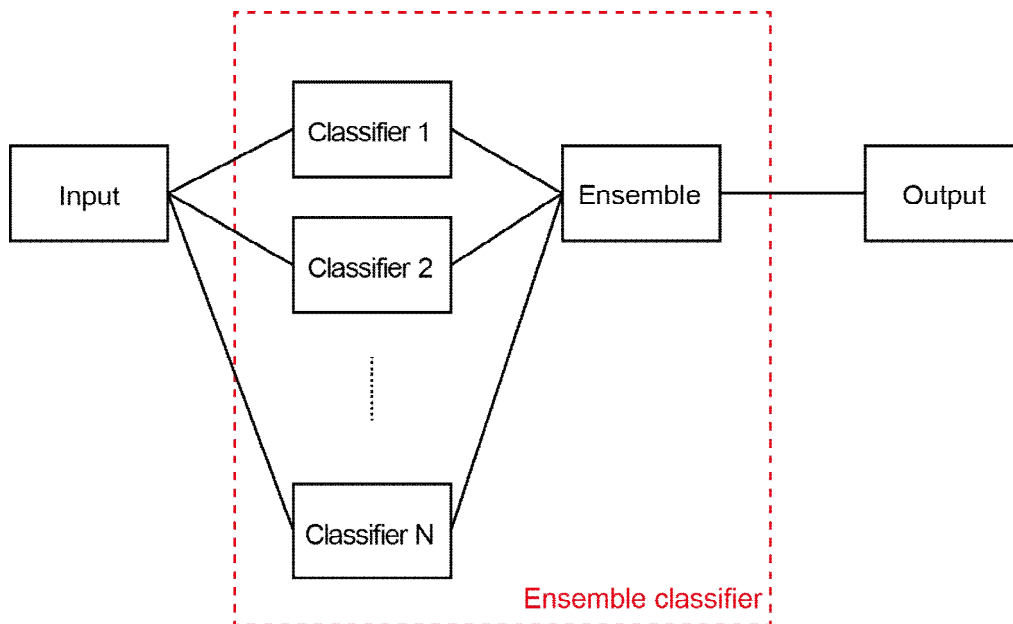


Figure 2-3. Example of ensemble model.

(4) Boosting

부스팅(boosting)은 1990년 Robert Schapire가 제안한 방법으로 순차적 약한 학습기(sequential weak learner)들을 여러 개 결합하여 예측 혹은 분류 성능을 높이는 알고리즘이다. 순차적 약한 학습기들을 이용하여 이전의 약한 학습기의 오차(error)를 다음 순서의 약한 학습기에 반영하여 모델을 개선해가는 방법이다. 기존의 학습 데이터에서 무작위 표본(random sampling)을 하고 1번 약한 학습기로 학습시킨다. 그 결과로 생긴 오차를 반영해 그다음 데이터 표본과 2번 약한 학습기에 학습시킨다(Freund et al., 1999; Schapire, 1999). 이 과정을 N 번 반복을 거치면 N 번 학습한 부스팅 모델이 된다(Figure 2-4). 부스팅 계열의 모델은 Gradient Boosting Machine, XGBoost, LightGBM, CatBoost가 있다.

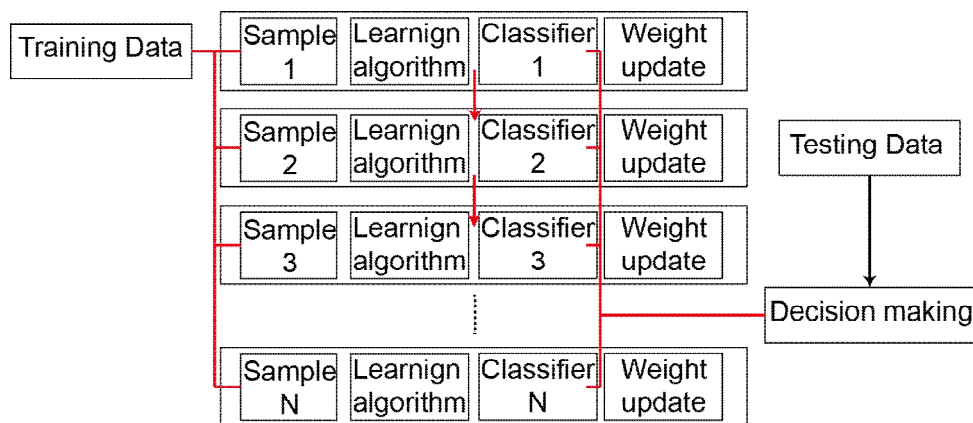


Figure 2-4. Example of boosting.

(5) GBM(Gradient Boost Machine)

순차적 약한 학습기들의 잔차(residual)를 줄이는 방향으로 결합하여 목적 함수(objective function)와 손실(loss)를 줄여나가는 것을 Gradient Boost Machine (GBM)이라 한다. GBM은 회귀와 분류를 수행할 수 있는 예측모형이며, 앙상블 방법론 중 부스팅 계열에 속하는 알고리즘이다. 표 형식(tabular format) 데이터에 대한 예측에서 엄청난 성능을 보여주고 머신러닝 알고리즘 중에서도 예측 성능이 높다고 알려져 이를 이용한 패키지들이 많다. 구체적인 과정은 Algorithm 1과 Figure

2-5와 같다(Natekin and Knoll, 2013; Wikipedia, 2022c).

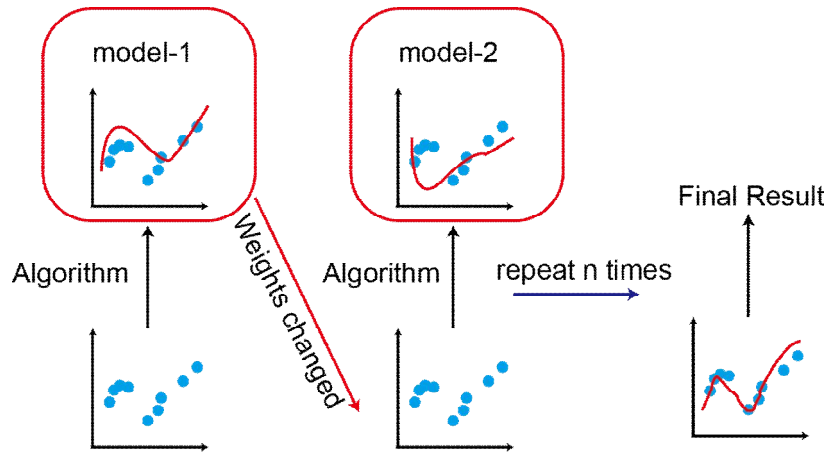


Figure 2-5. Example of gradient boost machine.

Algorithm 1. Gradient Boosting Machine

Input: training set $\{(x_i, y_i)\}_{i=1}^n$, a differentiable loss function $L(y, F(x))$, number of iterations M .

Algorithm:

1. Initialize model with a constant value:

$$F_0(x) = \arg \sum_{i=1}^n L(y_i, \gamma)$$

2. For $m = 1$ to M :

1. Compute so-called pseudo-residuals.

$$\gamma_{im} = - \left[\frac{\partial L(y_i, F(x_i))}{\partial F(x_i)} \right]_{F(x) = F_{m-1}(x)}, \text{ for } i = 1, \dots, n.$$

2. Fit a base learner (or weak learner, e.g. tree) closed under scaling $h_m(x)$ to pseudo-residuals, i.e.

train it using the training set $\{(x_i, \gamma_{im})\}_{i=1}^n$

3. Compute multiplier $\gamma_m = \arg \min_{\gamma} \sum_{i=1}^n L(y_i, F_{m-1}(x_i) + \gamma h_m(x_i))$

4. Update the model:

$$F_x(x) = F_{m-1}(x) + \gamma_m h_m(x)$$

3. Output $F_M(x)$

2.1.2 분류 모델의 평가 지표

우수한 모델이 갖추어야 할 요소들을 고려해 볼 때, 모델 평가는 여러 통계 모형을 비교 분석하는 것이 바람직하다. 이들 중 자료를 최적으로 설명하는 모형을 선택해야 한다. 최적의 모형을 선택하기 위해서는 여러 모델을 비교 평가해야 하고 선정된 최적의 모형이 다른 모형에 비하여 우수하다는 사실을 입증해야 한다. 모형을 평가하는 방법에는 예측력, 해석력, 효율성, 안정성 4가지 요소를 주로 평가하며 예측 문제에서 가장 중요한 고려사항은 예측력이다. 아무리 안정적이고 효율적이며 해석이 쉽더라도 실제 상황에 적용했을 때 빛나간 결과를 도출하는 경우 모델을 적용하는 의미가 없다. 따라서 모델의 평가는 예측을 위해 만든 모델이 임의의 모형보다 예측력이 우수한지 고려된 다른 모형 중 어느 모형이 가장 우수한 예측력을 보유하고 있는지를 비교 및 분석하는 과정이라 할 수 있다(Cho, 2017).

분류는 크게 이진 분류와 다중 분류로 나눌 수 있다. 이 분류의 평가 지표는 주로 2가지 방법을 사용한다. 첫째는 각 행 데이터가 양성(positive)인지 음성(negative)인지를 예측값으로 삼아 평가하는 지표이다. 둘째는 각 행 데이터가 양성(positive)일 확률을 예측값으로 삼아 평가하는 지표이다. 전자에는 혼동행렬(confusion matrix), 정확도, 오차율, 정밀도, 재현율, F1-Score 등이 있고 후자의 경우에는 AUC (Area Under the Curve)와 ROC (Receiver Operating Characteristic)가 있다.

(1) 혼동행렬

혼동행렬은 평가 지표는 아니지만, 양성인지 음성인지를 예측하는 정량적 수치의 지표로서 자주 활용되고 있다(Figure 2-6). 실제값과 예측값 모두 True인 빈도(True Positive), 실제값과 예측값 모두 False인 빈도(True Negative), 실제값은 False이나 True로 예측한 빈도(False Positive), 실제값은 True이나 False로 예측한 빈도(False Negative)로 구성된다. 이 산출 값들을 재가공하여 통계치로 사용한다. 이를 간단하게 정리하면 다음과 같다.

True Positive (TP) : 예측 모델은 양성이라고 진단한 반면, 실제로 양성인 경우

True Negative (TN) : 예측 모델은 음성이라고 진단한 반면, 실제로도 음성인 경우
 False Positive (FP) : 예측 모델은 양성이라고 진단한 반면, 실제로 음성인 경우
 False Negative (FN) : 예측 모델은 음성이라고 예측한 반면, 실제로 양성인 경우

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

Figure 2-6. Example of confusion matrix.

(2) 정확도와 오차율

정확도(accuracy)는 예측이 정확한 비율, 오차율(error rate)은 예측이 잘못된 비율을 나타내는 지표로 식 (2.3)과 (2.4)로 표현 된다. 정확하게 예측한 행 데이터 수를 전체 행 데이터 수로 나누어 구할 수 있는 직관적으로 이해하기 쉬운 평가 지표다. 불균형 데이터의 경우 모델의 성능을 평가하기가 어려워 많이 사용하지는 않는다.

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.3}$$

$$error\ rate = 1 - accuracy \tag{2.4}$$

(3) 정밀도와 재현율

정밀도(precision)는 양성으로 예측값 중에 실제값도 양성일 비율, 재현율(recall)은 실제값이 양성인 것 중에 예측값이 양성일 비율로 식 (2.5)와 (2.6)으로 표현할 수 있다. 정밀도와 재현율은 어느 한쪽의 값을 높이려 할 때 다른 쪽의 값은 낮아지는 트레이드 오프(Trade-off) 관계다. 반대로 어느 한쪽의 지표를 무시하면 다른 쪽 지표를 1에 가깝게 할 수 있다. 따라서 둘 중 하나만을 모델 성능의 지표로 삼는 일은 없어야 한다. 잘못된 예측(오답)을 줄이고 싶다면 정밀도를 중시하고, 실제 양성인 데이터를 양성으로 올바르게 예측하고 싶다면 재현율을 중요시해야 한다.

$$precision = \frac{TP}{TP + FP} \quad (2.5)$$

$$recall = \frac{TP}{TP + FN} \quad (2.6)$$

(4) F1-Score

F1-Score는 정밀도와 재현율의 조화평균으로 계산되는 지표로 식 (2.7)과 같다. F1-Score는 데이터 타겟이 불균형 구조일 때, 모델 성능을 정확하게 평가할 수 있으며 성능을 하나의 숫자로 표시할 수 있다.

$$F1\ Score = \frac{2}{\frac{1}{Precision} + \frac{1}{recall}} = 2 \left(\frac{Precision \times recall}{Precision + recall} \right) \quad (2.7)$$

(5) AUC-ROC Curve

대표적인 머신러닝을 평가하는 지표인 AUC와 ROC 곡선은 여러 절단값의 변화를 보면서 분류기의 성능 변화를 시각적으로 확인할 수 있다. ROC 곡선은 모든 임계값에서 분류 모델의 성능을 보여주는 그래프이다. ROC 곡선은 민감도가 y축, 1-specificity가 x축에 위치하며, 그래프의 좌상단에 붙어 있을수록 더 좋은 분류기를 의미한다. Figure 2-7의 노란색 ROC 곡선이 녹색 ROC보다 더 좋은 분류기로

평가될 수 있다. AUC는 ROC 곡선 아래의 영역이며 일반적으로 AUC 통계량 값이 클수록 예측력이 우수한 분류기로 평가하며, Figure 2-7의 파란색 AUC가 빨간색 AUC보다 높은 예측 성능 모델을 의미한다.

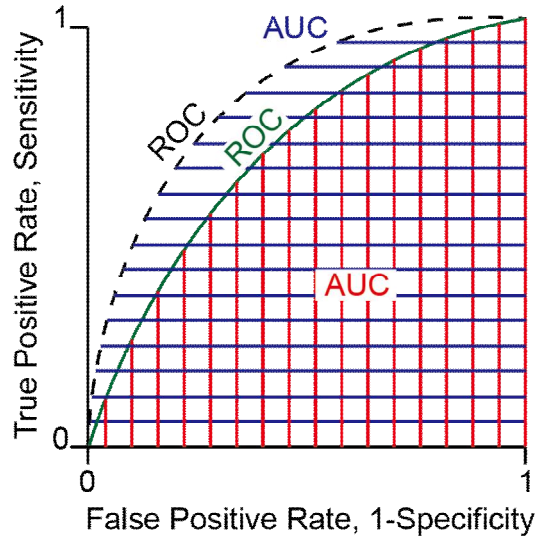


Figure 2-7. Example of AUC-ROC curve.

2.1.3 초매개변수 최적화

초매개변수(hyper-parameters)란 가중치(weight)와 같이 모델이 스스로 설정 및 갱신하는 변수가 아닌 사람이 직접 설정해야 하는 매개 변수(parameter)를 말한다. 인공지능망에서는 뉴런의 수, 배치의 크기, 학습률, 규제 등을 예로 들 수 있다. 모델은 이러한 초매개변수의 값에 따라 그 성능이 크게 좌우되기도 한다. 초매개변수 최적화(hyperparameter optimization)는 모델의 성능을 최대로 향상시킨 초매개변수를 탐색하는 것을 지칭하며, 크게 2가지로 구분할 수 있다. 초매개변수의 대략적인 범위를 설정한 후, 모델의 정확도를 평가 기준으로 사용자가 직접 최적값의 존재 범위를 조금씩 줄여나가는 방식인 수동 탐색(manual search)과 일정 범위를 프로그램이 자동으로 탐색하는 자동 탐색으로 분류할 수 있다. 초매개변수 자동 탐색은 주로 그리드 탐색(grid search), 무작위 탐색(random search), 그리고 베이지안 최적화(bayesian optimization)가 사용된다(Snoek et al., 2012).

그리드 탐색은 탐색 대상이 되는 특정 구간 내의 초매개변수 후보 값들을 일정한 간격으로 선정하여 가장 높은 성능을 발휘했던 초매개변수의 값을 선정하는 방법이다. 예를 들면, Figure 2-8 (a)는 초매개변수 A와 B를 그리드 탐색을 이용해 초매개변수를 찾는 과정을 나타낸다. X축, Y축 상단의 빨간색 선과 주황색 선은 각 매개변수의 성능을 나타낸 그래프이고 가장 높은 지점이 최적의 값이다. 따라서 매개변수 A는 매개변수 B 보다 상대적으로 중요한 매개변수이다. 매개변수 A와 B 모두 0~20을 5 간격으로 탐색 구간을 선정하여 총 16번 최적값 탐색한 결과, 매개변수 A에 대해서는 모델의 최고 성능을 수행할 수 있는 값을 탐색하지 못하였다. 그리드 탐색은 앞선 수동 탐색과 비교하면 좀 더 균등한 전역 탐색이 가능하다는 장점이 있다. 반면, 탐색 대상의 초매개변수 개수를 한 번에 여러 종류로 가져갈수록 전체 탐색 시간이 기하급수적으로 증가한다는 단점도 있다.

무작위 탐색은 그리드 탐색과 유사하나 탐색 대상 구간 내의 초매개변수 후보 값들을 무작위 샘플링 한다는 점에서 크게 다르다. Figure 2-8 (b)는 그리드 탐색과 같은 매개변수와 같은 구간에 대해 무작위 탐색 총 6회를 실시하였다. 무작위 탐색의 초매개변수 탐색 결과 총 16번 탐색한 그리드 탐색과 비슷한 모델성능을 나타냈다. 무작위 탐색은 그리드 탐색에 비해 불필요한 반복 수행 횟수를 대폭 줄이고, 동시에 정해진 간격 사이에 위치한 값들에 대해서도 확률적으로 탐색이 가능하여 최적의 초매개변수를 더 빨리 찾을 수 있는 반면, 정확도가 낮을 수 있다는

단점도 있다.

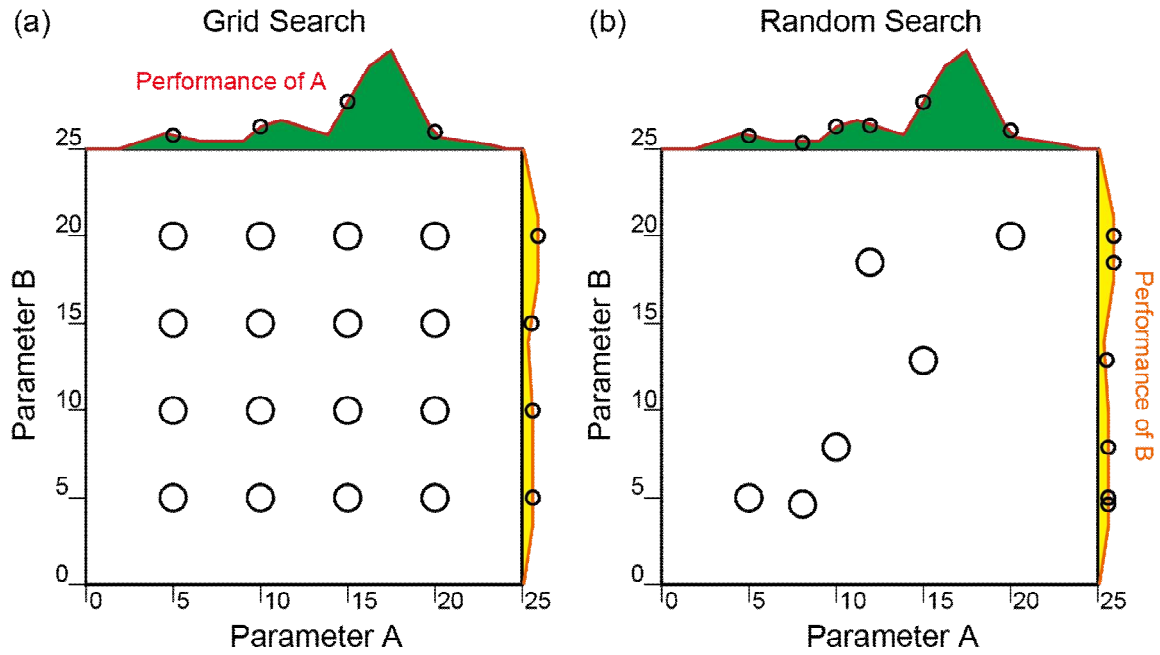


Figure 2-8. Schematic diagram of grid search and random search.

베이지안 최적화(bayesian optimization)는 어느 입력값 x 를 받는 미지의 목적함수(objective function) f 를 상정하여 그 함수값 $f(x)$ 를 최대로 만드는 최적해 x^* 를 찾는 것으로 식 (2.13)과 같다. 베이지안 최적화는 Surrogate model과 Acquisition function로 구성되었다. Surrogate model은 가우스 과정을 기반으로 현재까지 조사된 입력값과 함수값 점들을 바탕으로 미지의 목적함수 형태에 대한 확률적인 추정을 수행하는 모델이다. Acquisition function은 목적함수에 대한 현재까지의 확률적 추정 결과를 바탕으로 최적 입력값 x^* 를 찾는데 있어 가장 유용할 만한 다음 입력값 후보 x_{t+1} 을 추천해주는 함수이다. 쉽게 말하면 Surrogate model은 지금까지 확보된 데이터와 평가지표의 숨겨진 관계를 모델링하며 Acquisition function은 Surrogate model을 활용해 다음 탐색 지점을 결정하는 함수이다(Snoek et al., 2012; Frazier, 2018; Wikipedia, 2022a).

$$\begin{aligned}
 x^* &= \operatorname{argmax}_{x \in A} f(x) \\
 &= \operatorname{argmax} \text{Acquisition function}
 \end{aligned}
 \tag{2.13}$$

Figure 2-9은 학습률에 대하여 베이지안 최적화를 이용한 예시이다. 실선은 목적 함수(unknown black box function)를 나타내고, 점선은 지금까지 관측한 데이터를 바탕으로 예측한 추정함수(estimated function)이다. 점선 주변의 음영 영역은 목적함수 $f(x)$ 가 존재할만한 confidence bound (variance of function)을 의미한다. EI(x) 그래프는 Acquisition function을 나타내며, 다음 차례 입력값 후보를 추천해 준다. 획득함수(acquisition function) 값이 컸던 지점의 함수값을 관측하고 추정값(estimation)을 최신화(update)한다. 관측한 지점 중 최적 지점(best point)을 $\text{argmax } f(x)$ 로 선택한다(Snoek et al., 2012; Frazier, 2018; Wikipedia, 2022a).

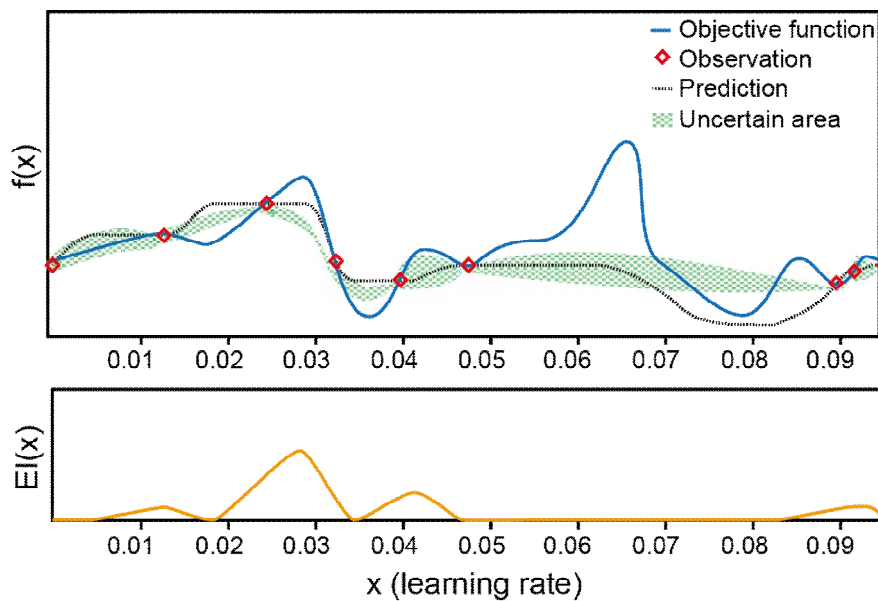


Figure 2-9. Example of Bayesian optimization for learning rate parameter.

2.2 XGBoost

2.2.1 XGBoost 이론적 배경

XGBoost는 기존 Gradient Boosting 알고리즘에 CART (Classification And Regression Tree) 모델을 사용하고 병렬처리를 가능하게 하여 데이터마이닝을 활용한 다양한 문제를 해결하는데 이용되는 방법이다(Chen and Guestrin, 2016; DSBA, 2020a; Yoon, 2020; An, 2021).

XGBoost에서 사용하는 목적 함수는 CART 알고리즘을 기반한다. 범주형 및 연속형 데이터 모두 학습이 가능한 CART의 계산과정은 일반적인 의사결정나무와 다르다. 의사결정나무(decision tree)는 잎 하나에 대해 결정값을 갖지만 CART는 모든 잎이 최종 점수와 연관되어 있다. 따라서 일반적인 의사결정나무는 분류의 정확도에만 초점을 맞추지만, CART는 트리의 모든 잎이 최종 점수와 연관되어 있어 같은 분류 결과를 갖는 모형들의 우위를 점수로 비교할 수 있다. Figure 2-10(a)은 9개 입력데이터에 형태와 색을 기준으로 노드를 나누고, 임의의 값이 할당된다. 하나의 트리의 결과는 예측 모델로 적합하지 않기 때문에, 여러 개의 트리를 사용해 각 트리의 결과의 합을 예측값으로 사용한다. 따라서, Figure 2-10(b)와 (c)로 분류한 결과를 종합하면 Figure 2-10(d)와 같다. 세모는 -1, 원은 3.5, 네모는 0.5의 예측값을 가지게 된 것으로 나타난다. 그 결과를 수식으로 일반화시키면 식 (2.16)과 같다(Chen and Guestrin, 2016; DSBA, 2020a; Yoon, 2020; An, 2021).

$$\hat{y}_i = \sum_{k=1}^K f_k(x_i), f_k \in F \quad (2.16)$$

여기서, \hat{y}_i 은 예측 점수, k 는 트리 개수, F 는 CART 공간, f 는 F 공간 함수를 나타낸다. 트리 앙상블에 대한 목적함수를 나타내는 식 (2.17)에서 $\sum_i^n l(y_i, \hat{y}_i)$ 은 학습 손실의 합, $\sum_{k=1}^K \Omega(f_k)$ 은 트리 복잡도의 합으로 나타난다(Chen and Guestrin, 2016; DSBA, 2020a; Yoon, 2020; An, 2021).

$$obj(\theta) = \sum_i^n l(y_i, \hat{y}_i) + \sum_{k=1}^K \Omega(f_k) \quad (2.17)$$

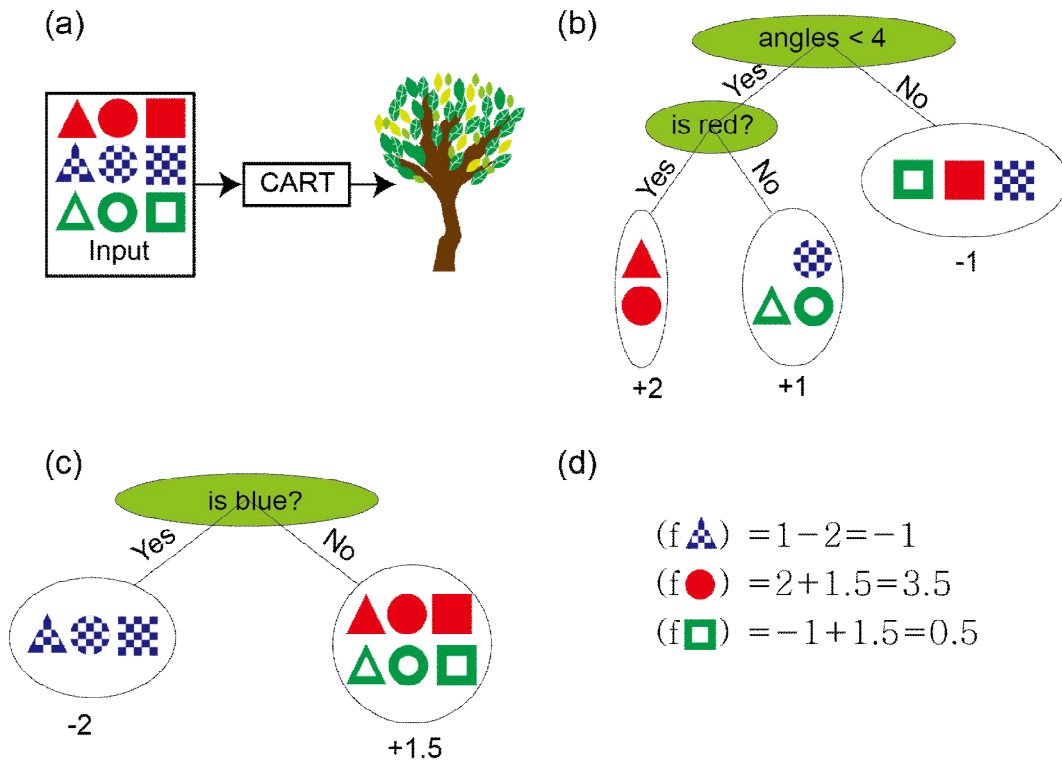


Figure 2-10. Example of tree ensemble model (a) flow chart of tree based model, (b) first weak learner, (c) second weak learner, (d) prediction score in each leaf.

XGBoost는 첨가(additive) 훈련방식을 사용한다. 첫 단계에서 예측값을 0으로 설정하고 t 단계의 예측값은 Boosting 원리에 따라 식 (2.18)과 같이 표현할 수 있다 (Chen and Guestrin, 2016; DSBA, 2020a; Yoon, 2020; An, 2021).

$$\begin{aligned} \hat{y}_i^{(0)} &= 0 \\ \hat{y}_i^{(1)} &= f_1(x_i) = \hat{y}_i^{(0)} + f_1(x_i) \\ \hat{y}_i^{(2)} &= f_1(x_i) + f_2(x_i) = \hat{y}_i^{(1)} + f_2(x_i) \\ &\dots \\ \hat{y}_i^{(t)} &= \sum_{k=1}^t f_k(x_i) = \hat{y}_i^{(t-1)} + f_t(x_i) \end{aligned} \quad (2.18)$$

t 단계에서 훈련시킬 목적함수는 식 (2.19)와 같이 정의할 수 있다(Chen and Guestrin, 2016; DSBA, 2020a; Yoon, 2020; An, 2021).

$$obj(t) = \sum_{i=1}^n l(y_i, \hat{y}_i^{(t)}) + \sum_{i=1}^t \Omega(f_i) = \sum_{i=1}^n l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(f_i) + constant \quad (2.19)$$

Boosting은 t가 지날수록 모형의 오차를 줄이는 과정이므로 손실함수가 평균제곱오차 (Mean Squared Error, MSE)면 식 (2.19)는 식 (2.20)로 나타낼 수 있다(Chen and Guestrin, 2016; DSBA, 2020a; Yoon, 2020; An, 2021).

$$\begin{aligned}
 obj(t) &= \sum_{i=1}^n (y - (\hat{y}_i^{(t-1)} + f_t(x_i)))^2 + \Omega(f_i) + constant & (2.20) \\
 &= \sum_{i=1}^n [2(\hat{y}_i^{(t-1)} - y_i)f_t(x_i) + f_t(x_i)^2] + \Omega(f_i) + constant
 \end{aligned}$$

목적함수의 복잡성을 해소하기 위해 Taylor 급수를 활용하면 목적함수의 근사치는 식 (2.21)과 같이 표현할 수 있다. 여기서, g함수는 $g_i = \delta_{\hat{y}^{(t-1)}} l(y_i, \hat{y}^{(t-1)})$ 로 정의되고, h함수는 $h_i = \delta_{\hat{y}^{(t-1)}}^2 l(y_i, \hat{y}^{(t-1)})$ 로 표현된다(Chen and Guestrin, 2016; DSBA, 2020a; Yoon, 2020; An, 2021).

$$obj(t) \simeq \sum_{i=1}^n [l(y_i, \hat{y}_i^{(t-1)}) + g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i)] + \Omega(f_i) + constant \quad (2.21)$$

기존 수식에서 상수를 제거하면 미분 형태의 계산만으로 손실함수를 재정의할 수 있으며 새로운 트리에 대한 학습을 최적화할 수 있는 t 단계의 최종 목표 함수 식인 식 (2.22)와 같이 표현할 수 있다(Chen and Guestrin, 2016; DSBA, 2020a; Yoon, 2020; An, 2021).

$$obj(t) \simeq \sum_{i=1}^n [g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i)] + \Omega(f_i) \quad (2.22)$$

g함수와 h함수는 t번째 이전 단계의 미분이므로 무한히 전개했을 때 수렴하는 형태가 된다. XGBoost는 g함수와 h함수에 대한 모든 근사식을 수정하여 다양한 손실함수를 활용할 수 있으며, 트리함수 $f(x)$ 는 식 (2.23)과 같이 정의된다(Chen and Guestrin, 2016; DSBA, 2020a; Yoon, 2020; An, 2021).

$$f_t(x) = w_{q(x)}, w \in R^T, q: R^d \rightarrow 1, 2, \dots, T \quad (2.23)$$

여기서 w 는 트리 잎의 가중치, q 는 트리 잎의 할당 함수, T 는 마디 개수를 나타낸다. 트리 한 개의 복잡도를 정의하면 식 (2.24)와 같다(Chen and Guestrin, 2016; DSBA, 2020a; Yoon, 2020; An, 2021).

$$\Omega(f_t) = \gamma T + \frac{1}{2} \lambda \sum_{j=1}^T w_j^2 \quad (2.24)$$

j번째 잎에 있는 인스턴스 구성을 $I_j = (i \mid q(x_i) = j)$ 로 정의하고, 목적함수를 다시 정리하면 식 (2.25)와 같다(Chen and Guestrin, 2016; DSBA, 2020a; Yoon, 2020; An, 2021).

$$\begin{aligned} Obj^{(t)} &\simeq \sum_{i=1}^n [g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i)] + \Omega(f_t) \\ &\simeq \sum_{i=1}^n [g_i w_{q(x_i)} + \frac{1}{2} h_i f_t^2(x_i)] + \gamma T + \frac{1}{2} \lambda \sum_{j=1}^T w_j^2 \\ &\simeq \sum_{j=1}^T [(\sum_{i \in I_j} g_i) w_j + \frac{1}{2} (\sum_{i \in I_j} h_i + \lambda) w_j^2] + \gamma T \end{aligned} \quad (2.25)$$

G를 g의 총합이라고 하면 $G_j = \sum_{i \in I_j} g_i$ 로 표현할 수 있고, H도 h의 총합으로 정의하면 $H_j = \sum_{i \in I_j} h_i$ 로 표현할 수 있다. 이들 함수를 식 (2.25)에 대입하면 다음과 같은 식 (2.26)을 얻을 수 있다(Chen and Guestrin, 2016; DSBA, 2020a; Yoon, 2020; An, 2021).

$$Obj^{(t)} = \sum_{j=1}^T [G_j w_j + \frac{1}{2}(H_j + \lambda)w_j^2] + \gamma T \quad (2.26)$$

여기서, 트리의 구조 $q(x)$ 는 고정되어 있고 j 에 최적의 가중치를 w_j^* 라고 하며 식 (2.27)로 표현할 수 있다(Chen and Guestrin, 2016; DSBA, 2020a; Yoon, 2020; An, 2021).

$$w_j^* = -\frac{G_j}{H_j + \lambda} \quad (2.27)$$

최적의 목적함수는 Obj^* 이며, 모형이 목적값을 통해 좋은 정도를 평가한다. 그 값이 낮을수록 더 좋은 트리 구조를 의미하며 구체적인 수식은 식 (2.28)과 같다(Chen and Guestrin, 2016; DSBA, 2020a; Yoon, 2020; An, 2021).

$$Obj = -\frac{1}{2} \sum_{j=1}^T \frac{G_j^2}{H_j + \lambda} + \gamma T \quad (2.28)$$

Figure 2-11은 각 입력데이터 별로 g_i, h_i 를 계산하고 이를 통해 해당 단계의 목적함수 값을 얻는 과정을 보여준다. 이 목적함수가 낮을수록 좋은 CART 모델이며, 각 노드의 G, H는 노드에 속하는 데이터들의 g와 h의 합으로 결정된다. 이상적인 방법으로는 가능한 모든 트리를 생성하고, 그 중에서 가장 최적의 트리를 선택하는 것이다. 깊이가 0인 트리에서 하나의 잎을 2개로 분리할 때, 식 (2.29)로 나타낼 수 있다. Gain은 식 (2.29)로 나누어진 2개의 노드에서 얻어진 최적값에 부모 노드의 최적값을 빼준 것이다. γ 는 하이퍼 파라미터이며, Gain이 γ 보다 작을 경우는 노드를 더 이상 분기하지 않는다(Chen and Guestrin, 2016; DSBA, 2020a; Yoon, 2020; An, 2021).

$$Gain = \frac{1}{2} \left[\frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} - \frac{(G_L + G_R)^2}{H_L + H_R + \lambda} \right] - \gamma \quad (2.29)$$

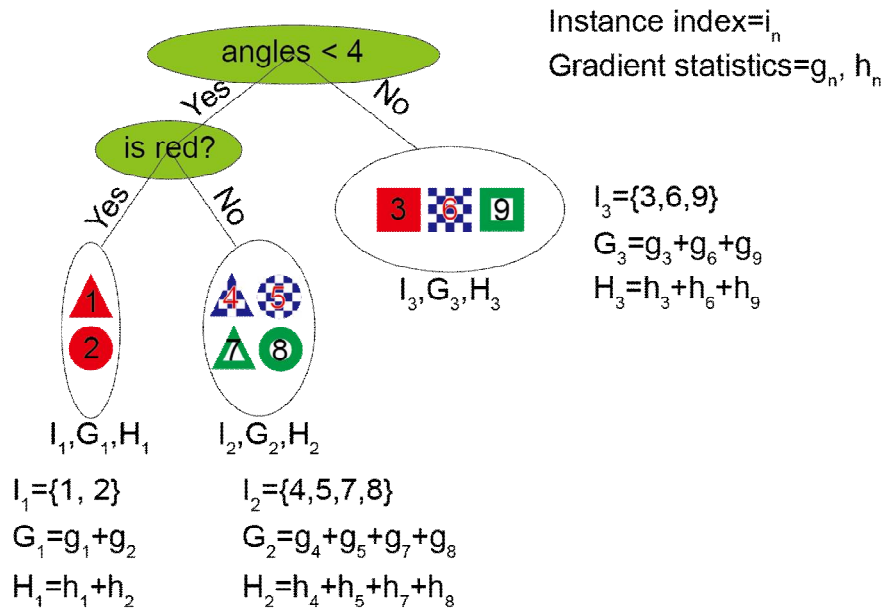


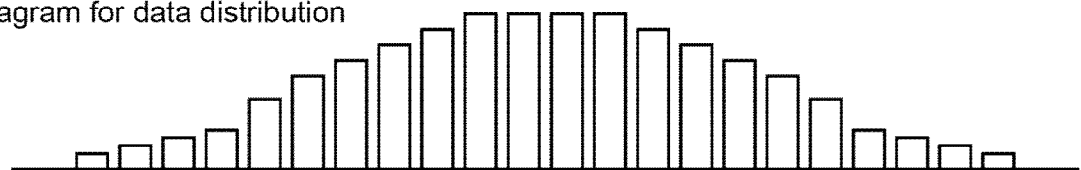
Figure 2-11. Example of structure score calculation in XGBoost.

트리 구조 학습의 핵심적인 문제 중 하나는 가장 좋은 분할 방법을 찾는 것이다. XGBoost에서는 트리분할 방법으로 Exact Greedy Algorithm과 Approximate Algorithm를 사용하고 각 방법의 절차는 Algorithm 2, 3과 같다. Exact Greedy Algorithm은 기존 부스팅 트리의 대부분에서 사용하는 알고리즘으로서 연속적인 feature들의 모든 구간을 분할하는 방법이다. Approximate algorithm은 분할 기준에 따라 global과 local로 나뉘고, Feature 분포의 백분위수(Percentiles)에 따라 후보 분할지점을 제시하고 그 지점으로 나뉜 버킷(Bucket)에 맵핑(mapping)을 하여 통계량을 집계해 가장 좋은 해를 찾는 방법이다. 입력데이터가 Figure 2-12(a)와 같이 분포할 때, greedy algorithm은 모든 구간에 대해 총 19번 분할 하여 연산을 수행한다(Figure 2-12(b)). Approximate algorithm의 경우는 지정한 초매개변수 수 만큼 버킷으로 분할 하여 각 구간에 대해 연산을 한다. Figure 2-12(c)와 Figure 2-12(d)는 approximate algorithm의 버킷 개수를 5로 지정했을 때 각 트리의 연산 횟수를 보여준다. Figure 2-12(c)의 최적 분할 지점(best split point)으로 분할되기 전 연산 횟수는 버킷의 수와 버킷 안의 분할 가능 개수를 곱해 총 15회이며, 최적 분할 지점으로 분할된 후의 연산 횟수는 14회임을 보여준다. Figure 2-12(d)는 트리 분할 전 분할과 연산 횟수는 15회이며, 트리 분할 후 연산 횟수는 20회임을 보여준다. 트리 당 Approximation algorithm의 경우, 분할 구간 기준이 나무이므로

나무의 깊이가 깊어져도 버킷의 수가 일정하지만, 분할 당 Approximation algorithm의 경우는 버킷의 분할 기준이 앞서므로 나무의 깊이가 깊어질수록 분할 구간과 연산 횟수도 증가한다(Chen and Guestrin, 2016; DSBA, 2020a).

Algorithm 4와 Figure 2-13은 데이터 내의 결측치를 다루는 방법으로서 결측치가 존재하는 경우 또는 '0'이 많은 때 One-Hot encoding 같은 feature engineering artifacts를 사용한 경우에 유용하다. Figure 3-13(d)에서 A와 C는 하나의 결측치가 각각 존재한다. 이를 해당 트리 모델에서 분류할 경우 전 모델에서 산정된 default 방향으로 분류된다(Chen and Guestrin, 2016; DSBA, 2020a).

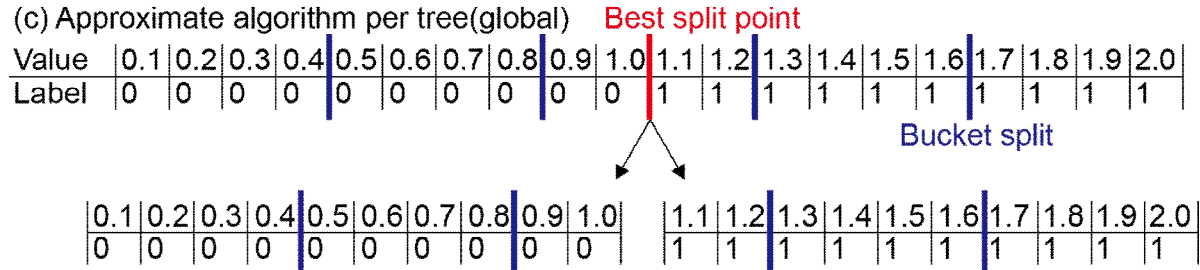
(a) Diagram for data distribution



(b) Greedy algorithm

Value	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	1.9	2.0	
Label	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1

(c) Approximate algorithm per tree(global)



(d) Approximate algorithm per split(local)

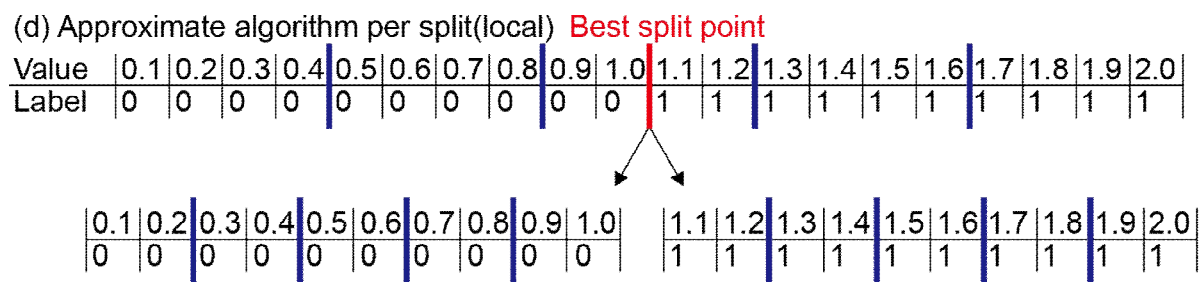


Figure 2-12. Examples of Exact Greedy and Approximate algorithm for split finding.

Algorithm 2. Exact greedy algorithm for split finding

Input : I, instance set of current node
 Input : d, feature dimension
 gain \leftarrow 0
 $G \leftarrow \sum_{i \in I} g_i, H \leftarrow \sum_{i \in I} h_i$
 for k = 1 to m do
 $G_L \leftarrow 0, H_L \leftarrow 0$
 for j in sorted(I, by x_{jk}) do
 $G_L \leftarrow G_L + g_j, H_L \leftarrow H_L + h_j$
 $G_R \leftarrow G - G_L, H_R \leftarrow H - H_L$
 score \leftarrow max(score, $\frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} - \frac{G^2}{H + \lambda}$)
 end
 end
 Output: Split with max score

Algorithm 3. Approximate algorithm for split finding

for k = 1 to m do
 Propose $S_k = \{s_{k1}, s_{k2}, s_{k3} \cdots s_{kl}\}$ by percentiles on feature k.
 Proposal can be done per tree(global), or per split(local)
 end
 for k = 1 to m do
 $G_{kv} \leftarrow \sum_{j \in \{j | s_{k,v} \geq x_{jk} > s_{k,v-1}\}} g_i$
 $H_{kv} \leftarrow \sum_{j \in \{j | s_{k,v} \geq x_{jk} > s_{k,v-1}\}} h_i$
 end
 Follow same step as in previous section to find max score only among proposed splits.

(a)

Value	1.5	0.3	0.3	1.2	0.1	0.7	0.9	0.9
Label	1	0	0	1	0	1	0	1

(b)

Value	0.1	0.3	0.7	0.9	1.2	1.5			
Label	0	0	1	1	1	1	0	0	0

(c)

Value					0.1	0.3	0.7	0.9	1.2	1.5
Label	0	0	0	0	0	0	1	1	1	1

Best split point

(d)

Name	Age	Gender
A	-	female
B	10	-
C	30	male

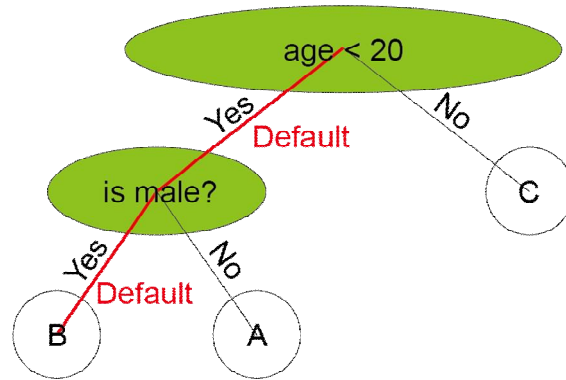


Figure 2-13. Example of Sparsity-Aware split finding (a) initial dataset, (b) missing value goto right in dataset, (c) missing value goto left in the dataset, (d) Tree structure with missing values for age and gender.

Algorithm 4. Sparsity-aware split finding

Input: I , instance set of current node

Input: $I_k = \{i \in I | x_{ix} \neq \text{missing}\}$

Input: d , feature dimension

also applies to the approximate setting, only collect statistics of non-missing entries into buckets

$gain \leftarrow 0$

$G \leftarrow \sum_{i \in I} g_i, H \leftarrow \sum_{i \in I} h_i$

for $k = 1$ to m do

 //enumerate missing value go to right

$G_L \leftarrow 0, H_L \leftarrow 0$

 for j in sorted (I_k , ascent order by x_{jk}) do

$G_L \leftarrow G_L + g_j, H_L \leftarrow H_L + h_j$

$G_R \leftarrow G - G_L, H_R \leftarrow H - H_L$

$score \leftarrow \max(score, \frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} - \frac{G^2}{H + \lambda})$

 end

 //enumerate missing value go to left

$G_R \leftarrow 0, H_R \leftarrow 0$

 for j in sorted (I_k , ascent order by x_{jk}) do

$G_R \leftarrow G_R + g_j, H_R \leftarrow H_R + h_j$

$G_L \leftarrow G - G_R, H_L \leftarrow H - H_R$

$score \leftarrow \max(score, \frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} - \frac{G^2}{H + \lambda})$

 end

end

Output: Split and default discretions with max gain

2.2.2 XGBoost 학습 파라미터

XGBoost를 사용하기 위해서는 3가지 유형의 파라미터를 설정해야 한다. 일반 매개변수(general paramter)는 일반적으로 트리 또는 선형 모델에서 부스팅을 수행하는데 사용하는 부스터와 관련된 파라미터이다. 부스터 매개변수(booster parameter)는 각 스텝에서 부스터(booster)를 가이드하는 역할을 수행한다. 학습 임무 수행 파라미터(learning task parameter)는 학습 시나리오를 결정한다. 각각 유형에 해당하는 주요 파라미터는 다음 Table 2-1과 같다.

Table 2-1. Parameters mainly used in XGBoost

Parameter	Description
booster	A parameter to set the tree based model of XGBoost.
colsample_bylevel	A family of parameters for subsampling of columns. The subsample ratio of coluns for each level.
colsample_bytree	A family of parameters for subsampling of columns. The subsample ratio of columns when constructing each tree.
gamma	Minimum loss reduction required to make a further partition on a leaf node of the tree.
learning_rate	Step size shrinkage used in update to prevents overfitting.
max_delta_step	Maximum delta step we allow each leaf output to be.
max_depth	Maximum depth of a tree.
min_child_weight	Minimum sum of instance weight needed in a child.
n_estimators	The number of weak learners.
reg_alpha	L1 regularization term on weights.
reg_lambda	L2 regularization term on weights.
scale_pos_weight	Control the balance of positive and negative weights, useful for unbalanced classes.
subsample	Subsample ratio of the training instances.
objective	Objective function. Specifying the learning task.
eval metric	Specifying the evaluation parameter about training.

2.3 LightGBM

2.3.1 LightGBM 이론적 배경

Microsoft에서 개발한 LightGBM은 Gradient boosting 알고리즘을 기존 알고리즘보다 가볍게 구현하여 학습시간을 획기적으로 단축시킨 알고리즘이다. 따라서 학습하는데 소요되는 시간이 다른 앙상블 알고리즘에 비해 적고, 메모리 사용량이 낮다. LightGBM은 잎 분할 방식을 사용한다(Figure 1-1(b)). 기존 의사결정트리 알고리즘들은 트리의 깊이를 균등하게 분할하는 방식을 사용하지만, LightGBM은 깊이를 고려하지 않고 무조건 최대손실 값을 가지는 노드를 분할하여 비대칭적인 트리가 구축된다. 따라서 기존에 트리의 깊이를 균등하게 분할하기 위해 소요되던 시간을 줄였다. 하지만, 다른 의사결정 트리기반의 알고리즘보다 과적합의 위험에 노출되며, 특히 10,000건 이하의 데이터에 적용할 경우 과적합의 위험이 크다(Ke et al., 2017; Al Daoud, 2019; Yoon, 2020; An, 2021).

LightGBM의 Histogram-based algorithm은 Algorithm 5와 Figure 2-14에서 볼 수 있듯이 연속적인 변수를 이산적 구간(bin)으로 나누고, 이 구간을 학습 과정에서 피쳐 히스토그램(Feature histogram)을 구성한다. 이 방법은 결정트리를 사용하여 입력공간 X^s 에서 기울기 공간 g 까지의 함수를 학습한다. n 개의 독립항등분포(i.i.d) 개체 $\{x_1, \dots, x_n\}$ 의 훈련셋을 가지고 있을 때, 각 x_i 는 공간 X^s 내 차원이 s 인 벡터다. 그래디언트 부스팅은 매회 반복마다 발생하는 모형 출력값에 대한 손실함수의 음의 기울기를 $\{g_1, \dots, g_n\}$ 로 정의한다. 이 모형은 가장 큰 정보획득(information gain)을 갖는 변수로 각 노드를 분할한다. 이 때 사용한 정보 획득은 다음과 같은 식 (2.30)으로 정의될 수 있다(Ke et al., 2017; Al Daoud, 2019; DSBA, 2020b; Yoon, 2020; An, 2021).

$$V_{j|o}(d) = \frac{1}{n_o} \left(\frac{\left(\sum_{x_i \in O: x_{ij} \leq d} g_i \right)^2}{n_{l|o}^j(d)} + \frac{\left(\sum_{x_i \in O: x_{ij} > d} g_i \right)^2}{n_{r|o}^j(d)} \right) \quad (2.30)$$

$$n_o = \sum I[x_i \in O], n_{l|o}^j(d) = \sum I[x_i \in O: x_{ij} \leq d], n_{r|o}^j(d) = \sum I[x_i \in O: x_{ij} > d]$$

여기서, O 는 트리 노드 안에 있는 훈련데이터 셋이고, d 는 노드, j 는 점 d 에서 분할하는 변수이다. 변수 j 에 대해 트리 알고리즘은 $d_j^* = \operatorname{argmax}_d V_j(d)$ 를 선택하고 최대 획득 $V_j(d_j^*)$ 를 계산한다. 다음으로 데이터를 변수 j^* 의 점 d_{j^*} 에 따라 왼쪽과 오른쪽 하위 노드로 분할한다. 하지만 이 방법은 모든 데이터 구간을 탐색해야 하므로 데이터셋이 큰 경우 효율적이지 못하다. 학습 데이터의 양을 줄이기 위해 가장 쉽게 생각할 수 있는 방법은 다운샘플링이다. 이 방법은 데이터 개체의 중요도 (weight)가 설정한 임계값을 넘지 못할 경우 데이터 개체들이 필터링 되는 과정을 말한다. 다운샘플링 외에도 중복요소가 많이 포함되어 있는 약한 피쳐들을 필터링 하는 방법도 있다. 실제 상황에서 사용되는 대용량의 데이터셋은 희소한(sparse) 성질을 갖는 확률이 높다. LightGBM은 데이터 개체의 중요도(weight)를 사용하지 않는 Gradient Boosting Decision Tree (GBDT)를 기반하므로 희소한 데이터셋은 제외시켜야 한다. 이를 위한 해결책은 앞서 설명한 데이터 개체 수를 줄이는 GOSS (Gradient-based One-Side Sampling)와 피쳐 수를 줄이는 EFB (Exclusive Feature Bundling)이다(Ke et al., 2017; Al Daoud, 2019; DSBA, 2020b).

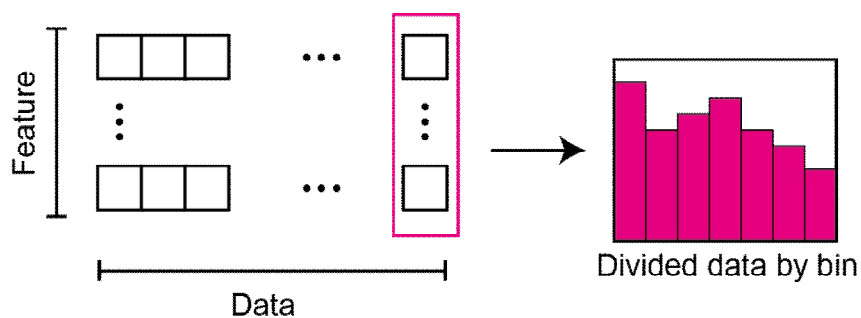


Figure 2-14. Example of Histogram-based algorithm.

Algorithm 5. Histogram-based Algorithm

Input: I : training data, d : max depth

Input: m : feature dimension

$nodeSet \leftarrow \{0\}$ ▷ tree nodes in current level

$rowSet \leftarrow \{0,1,2,\dots\}$ ▷ data indices in tree nodes

for $i = 1$ to d do

 for node in $nodeSet$ do

$usedRows \leftarrow rowSet[node]$

 for $k = 1$ to m do

$H \leftarrow new\ Histogram()$

 ▷ build histogram

 for j in $usedRows$ do

$bin \leftarrow I.f[k][j].bin$

$H[bin].y \leftarrow H[bin].y + I.y[j]$

$H[bin].n \leftarrow H[bin].n + 1$

 Find the best split on histogram H

 ...

 Update $rowSet$ and $nodeSet$ according to the best split points.

...

GBDT에서는 기울기가 데이터 개체의 중요도를 알려주는 역할을 수행한다. 각 데이터 개체의 기울기가 작으면 훈련 오차가 작다는 것을 의미하고 이는 학습이 잘되었다는 것을 뜻한다. GOSS는 데이터 개체들을 기울기의 절대값에 따라 정렬하고(Figure 2-15, Algorithm 6), 상위 a% 유지하고 작은 기울기를 갖는 데이터 개체는 b% 만큼 무작위 샘플링을 진행한다(Ke et al., 2017; DSBA, 2020b).

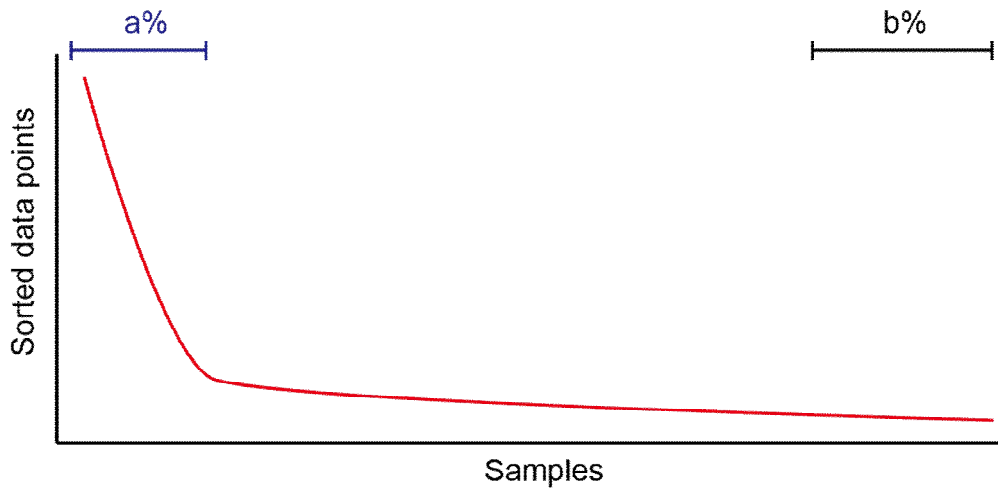


Figure 2-15. Example of Gradient-based One Side Sampling (GOSS).

Algorithm 6. Gradient-based One-Side Sampling
--

```

Input: I: training data, d: iterations
Input: a: sampling ratio of large gradient data
Input: b: sampling ratio of small gradient data
Input: loss: loss function, L: weak learner

models ← { }, fact ←  $\frac{1-a}{b}$ 

topN ← a × len(I), randN ← b × len(I)
for i = 1 to d do
    preds ← models.predict(I)
    g ← loss(I, preds), w ← {1, 1, ...}
    sorted ← GetSortedIndices(abs(g))
    topSet ← sorted[1:topN]
    randSet ← RandomPick(sorted[topN:len(I)], randN)
    usedSet ← topSet + randSet
    w[randSet] × ← fact ▷ Assign weight fact to the small gradient data
    newModel ← L(I[usedSet], -g[usedSet], w[usedSet])
    models.append(newModel)
    
```

EFB는 변수 개수를 줄이기 위해 최소한 변수 공간의 특성에 따라 상호배타적인 변수들을 묶는 기법이다. 이를 위해 Greedy bundling을 사용해 어떤 변수를 함께 묶을지 결정하고(Figure 2-16, Algorithm 7), Merge Exclusive Feature를 이용하여 어떻게 묶을지도 결정한다(Figure 2-17, Algorithm 8). 이를 위해 가장 적은 수의 배타적 묶음으로 변수를 나누는 문제는 LightGBM에서는 NP-hard라 한다. NP-hard는 NP (non-deterministic polynomial)에 속하는 모든 판정 문제를 다항시간에 다대일 환산할 수 있는 문제들의 집합이며, 복잡계 이론에서는 NP-hard라 불리는 문제들은 모든 경우의 수를 전부 확인해보는 방법 이외에는 정확한 답을 구하는 방법이 없는 것을 뜻한다. 따라서, 가능한 모든 변수들의 관계를 전부 확인해야 한다. 우선, $G=(V, E)$ 라는 임의의 그래프가 있다고 가정하면, G 의 발생 행렬 (Incidence matrix)의 행들이 문제의 모든 변수에 해당한다. 변수(feature)들을 꼭지점(vertices)으로 간주하고 두 변수가 상호배타적일 경우, 그들 사이에 변(edge)을 추가하는 방식으로 이루어진다. Figure 2-16(a)는 7개의 변수가 있을 때 변수 간의 상호 관계를 나타낸다. Figure 2-16(a)에서 충돌(conflict)은 가중치를 의미하고 충돌이 없는 관계는 non-zero value가 동시에 존재하여 상호배타적이지 않은 상태를 의미한다. 그래프 내의 꼭지점 차수에 따라 내림차순으로 변수를 정렬하고 상호배타적인 변수끼리 묶음을 실행하면 Figure 2-16(b) 경우 4개의 묶음(bundle)이 생성된다(Ke et al., 2017; DSBA, 2020b).

하나로 묶여진 변수들은 훈련 복잡도를 줄이기 위해 같이 묶인 변수를 병합하는 방법이 필요하다. 원래의 변수값이 변수 묶음에서 식별 가능한지 확인하는게 중요하다. 히스토그램 기반 알고리즘은 변수의 연속적인 값 대신 개별 구간을 저장하기 때문에 배타적 변수를 각기 다른 구간에 두고 변수 묶음을 구성할 수 있다. EFB는 변수의 원래 값에 기준이 되는 변수의 값을 더하는 오프셋(off-set)을 사용한다. Figure 2-17은 두 변수 parameter 1과 parameter 2를 병합하는 과정을 나타내고 있다. parameter 1이 기준일 경우, 서로 배타적일 때 parameter 2에 parameter 1의 최대값을 더해 묶음 변수(bundle parameter) 값을 취하게 된다. EFB는 많은 배타적 변수를 훨씬 적은 밀집 변수로 묶어 줄 수 있기 때문에 변수의 0 값에 대한 불필요한 계산을 효과적으로 피할 수 있다(Ke et al., 2017; DSBA, 2020b).

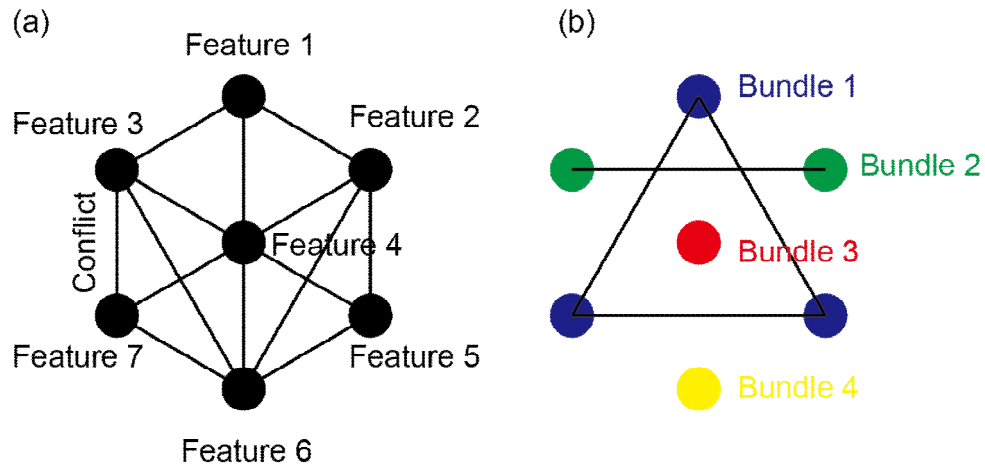


Figure 2-16. Example of Greedy Bundling (a) before bundling (b) after bundling.

Algorithm 7. Greedy Bundling

```

Input: F: features, K: max conflict count
Construct graph G
searchOrder ← G.sortByDegree()
bundles ← { }, bundlesConflict ← { }
for i in searchOrder do
    needNew ← True
    for j=1 to len(bundles) do
        cnt ← ConflictCnt(bundles[j], F[i])
        if cnt + bundlesConflict[i] ≤ K then
            bundles[j].add(F[i]), needNew ← False
            break
    if needNew then
        Add F[i] as a new bundle to bundles
Output: bundles
    
```

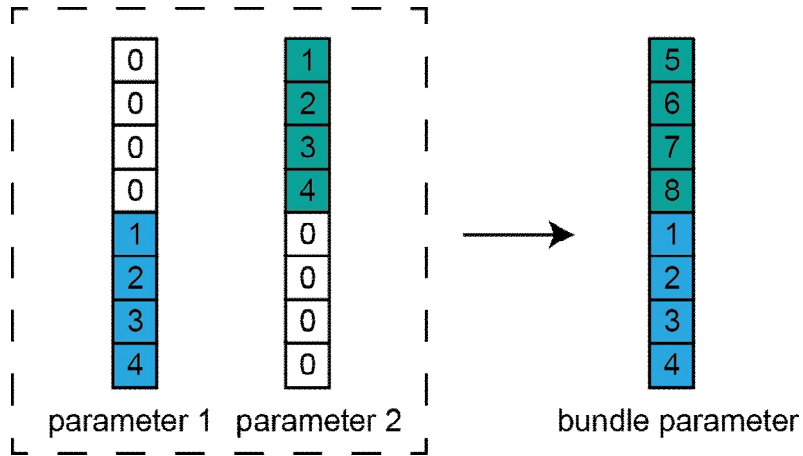



Figure 2-17. Example of Merge Exclusive Feature.

Algorithm 8. Merge Exclusive Features

```

Input: numData: number of data
Input: F; One bundle of exclusive features
binRange ← {0}, totalBin ← 0
for f in F do
    totalBin += f.numBin
    binRanges.append(totalBin)
newBin ← new Bin(numData)
for i=1 to num Data do
    newBin[i] ← 0
    for j=1 to len(F) do
        if F[j].bin[i] ≠ 0 then
            newBin[i] ← F[j].bin[i] + binRanges[j]
Output: newBin, binRanges
    
```

2.3.2 LightGBM 학습 파라미터

LightGBM의 파라미터는 목적에 따라 핵심 파라미터(core parameters), 학습 억제 파라미터(learning control parameters), 지표 파라미터(metric parameter) 등으로 분류할 수 있다. 학습에 사용되는 주요 파라미터는 Table 2-2와 같다.

Table 2-2. Parameters mainly used in LightGBM

Parameter	Description
booster	A parameter to set the tree based model of LightGBM.
num_threads	Parameter to set the thread to be processed in parallel processing.
learning_rate	Parameter that controls the learning rate of the model.
num_iterations	Parameter that specifies the number of boosts.
max_depth	Parameter that controls the depth of the tree.
num_leaves	Parameter that controls the maximum number of leaves a tree can have.
bagging_fraction	A parameter that increases diversity by extracting a part of data.
bagging_freq	Parameter that regulates data updates in an iterative process.
feature_fraction	Parameter that control the diversity of each column through sampling.
lambda_l1	L1 regularization term on weights.
lambda_l2	L2 regularization term on weights.
scale_pos_weight	Control the balance of positive and negative weight, useful for unbalanced classes.
objective	Objective function. Specifying the learning task.
eval_metric	Specifying the evaluation parameter about training.

2.4 CatBoost

2.4.1 CatBoost 이론적 배경

CatBoost (Categorical Boosting)는 Yandex에서 개발한 알고리즘으로 범주형 데이터 처리에 유용하다고 알려져 있다. Yandex에 의하면 기존의 Gradient Boosting을 사용하는 모델에는 크게 2가지 문제가 있다. 첫째, 범주형 데이터의 처리 문제다. 이전까지 범주형 데이터는 새로운 이진변수를 추가하거나 수치 형태로 변환하여 사용하였다. 범주형 데이터가 많아질수록 feature의 수가 급격하게 증가하고, 범주형 데이터의 target statistics (TS)를 추정하여 사용하기 때문에 타겟(target) 정보들이 사전에 유출되는 문제점인 타겟 유출(target leakage)이 발생하게 된다. 둘째로는, 잔차(residual)를 이용하여 손실함수를 사용하는 gradient boosting의 특성상 타겟을 활용하여 생기는 타겟 유출(target leakage)로 인해 훈련과 시험데이터셋의 예측값의 분포에 차이가 생기게 되며, 과적합 또는 부정확한 예측을 발생시키는 예측 변화(prediction shift)가 나타난다. CatBoost는 이런 타겟 유출과 예측 변화로 인한 과적합 문제를 Ordered TS와 Ordered boosting을 이용하여 해결하였다(Prokhorenkova et al., 2017; An, 2021).

먼저 범주형 변수들은 서로 겹치지 않은 그룹으로 구성된 데이터다. 학습을 위해서는 범주형 데이터를 숫자형으로 바꿔야 하며, 대표적인 방법으로는 one-hot encoding을 들 수 있다. 하지만 범주형 데이터가 많을 경우 one-hot encoding 사용시 데이터셋의 크기가 커지게 된다. 이를 방지하기 위해 one-hot encoding 수행 전 범주를 군집화하여 그 수를 줄이거나, 새로운 숫자형 데이터로 변환시킨다. 후자의 경우를 target statistic이라 하며, 식 (2.31)과 같이 \hat{x}_k^i 는 x_i 가 x_k^i 일 때 y 의 예측값으로 추정한다(Prokhorenkova et al., 2017; DSBA, 2020c).

$$\hat{x}_k^i \approx \mathbb{E}(y \mid x_i = x_k^i) \quad (2.31)$$

보편적으로 사용했던 방식은 식 (2.32)와 같은 Greedy TS이다. 여기서, \mathbb{I} 는 1차원 이진변수 공간을 나타낸다. k 번째 인스턴스(instance) x_k 의 categorical feature i

를 추정하는 \hat{x}_k^i 은 x_k 와 같은 카테고리에 속하는 모든 인스턴스들의 타겟 y 의 예측값으로 추정한다. 여기서, a 는 파라미터로서 $a > 0$ 조건을 만족해야하고, p 는 데이터셋 타겟값의 평균이다. a 와 p 값을 사용하는 이유는 빈도가 낮은 범주의 경우, 발생하는 데이터 노이즈를 최소화하기 위해서다(Prokhorenkova et al., 2017; DSBA, 2020c).

$$\hat{x}_k^i = \frac{\sum_{j=1}^n \mathbb{I}_{\{x_j^i = x_k^i\}} \cdot y_j + ap}{\sum_{j=1}^n \mathbb{I}_{\{x_j^i = x_k^i\}} + a} \quad (2.32)$$

Greedy TS도 범주형 변수를 가지고 있는 k 번째 객체를 수치형 변수로 바꾸는 과정에서 해당하는 객체의 정보 y_k 가 사용되어 타겟 유출이 발생한다. 또한 범주형 변수 i 에 대해 모든 값이 다르고, 각 범주 A 에 대해 확률 $P(y = 1 \mid x^i = A) = 0.5$ 일 경우 훈련과 시험데이터셋 사이에서 \hat{x}_k^i 의 분포가 상이한 conditional shift가 발생할 수도 있다. 이를 방지하기 위해 훈련데이터셋과 시험데이터셋에서 산출된 TS 값의 평균이 서로 같아야 한다는 Property 1을 제안되었다(Prokhorenkova et al., 2017; DSBA, 2020c).

$$\text{Property 1: } \mathbb{E}(\hat{x}^i \mid y = v) = \mathbb{E}(x^i \mid y_k = v)$$

Conditional shift를 방지하기 위한 일반적인 방법은 k 번째 데이터 x_k 의 TS를 계산할 때 해당 데이터를 제외하고 TS를 산출하는 것이다. 즉 데이터셋 $D_k \subset D$ 에 대하여 x_k 를 식 (2.33)와 같이 TS를 산출하는 것이다(Prokhorenkova et al., 2017; DSBA, 2020c).

$$\hat{x}_k^i = \frac{\sum_{x_j \in D_k} \mathbb{I}_{\{x_j^i = x_k^i\}} \cdot y_j + ap}{\sum_{x_j \in D_k} \mathbb{I}_{\{x_j^i = x_k^i\}} + a} \quad (2.33)$$

Conditional shift를 방지하기 위한 또 다른 방법으로는 Holdout TS가 있다. Holdout TS는 훈련데이터셋을 두 부분으로 나누어 하나(\widehat{D}_0)는 TS계산(D_k), 나머지 하나(\widehat{D}_1)는 학습에 활용하는 방법으로 식 (2.34)와 같다(Prokhorenkova et al., 2017; DSBA, 2020c).

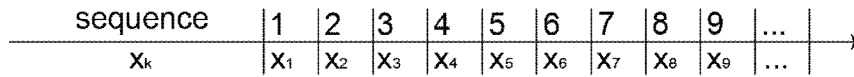
$$\begin{aligned}
 D &= \widehat{D}_0 \cup \widehat{D}_1 \\
 D_k &= \widehat{D}_0
 \end{aligned}
 \tag{2.34}$$

하지만 이 방법의 경우 property 1은 만족하지만, 학습과 TS 계산에 사용하는 데이터가 분리되어 데이터양이 줄어든다는 단점이 있다. 이를 극복하기 위해 Property 2가 제안되었다(Prokhorenkova et al., 2017; DSBA, 2020c).

Property 2: Effective usage of all training data for calculating TS features and for learning a model

CatBoost는 property 1과 2, 식 (2.31)과 (2.32)를 모두 만족하는 방법으로 Ordered TS를 제안한다. 데이터셋에 임의시간(random permutation, σ)을 부여해 k 번째 데이터의 TS를 계산할 때, k-1번까지 데이터의 타겟값(target value)만을 사용하는 것이다. Figure 2-18(a)는 임의시간 적용 전, Figure 2-18(b)는 적용 후를 나타낸다. 임의시간 7번째의 x_k 를 구하기 위해서 이전시간의 데이터셋 D_k 를 이용하며, 임의시간 n까지 이를 반복하여 Figure 2-18(b)의 boosting 단계에서의 TS를 구한다. 종합하면, Ordered TS는 observed history를 이용하는 방식으로 현 시점을 기준으로 과거 데이터로 TS를 추정하기 위해서 무작위 순열이라는 인공적 시간이 도입된다. 한 boosting 단계에서 하나의 무작위 순열만으로 TS를 추정하며, 각 boosting 단계마다 다른 무작위 순열을 활용한다(Prokhorenkova et al., 2017; DSBA, 2020c).

(a) Before random permutation



(b) After random permutation

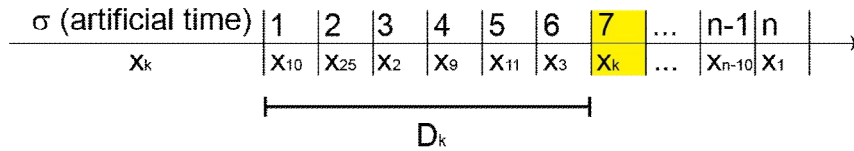


Figure 2-18. Ordered target statistics (a) before random permutation, (b) after random permutation.

예측 변화의 원인이 되는 타겟 유출을 방지하기 위해서는 트리를 학습하는 각 스텝마다 다른 데이터셋을 활용해야 한다. CatBoost에서 제안한 Ordered boosting은 임의의 순열을 만들어 순차적으로 잔차를 계산하여 트리를 학습하면서 타겟 유출을 방지하는 것이다(Algorithm 9). Figure 2-19는 순차적 부스팅을 나타낸 것으로 Figure 2-19(a)는 무작위 순열 전 데이터셋을 나타낸다. Figure 2-19(b)는 무작위 순열 후 1번째 잔차를 구하는 단계이고, Figure 2-19(c)는 2번째 모델에 대한 잔차를 구하는 단계이다. 이 과정을 계속 반복하여 Figure 2-19(d)는 7번째 모델의 잔차를 구하고 모델을 업데이트하는 단계를 나타낸다. 하지만, 이러한 방식은 데이터 개수만큼의 서로 다른 학습 모델을 필요로 하고 이는 모델의 복잡도와 메모리 요구량을 상승시킨다. CatBoost의 Ordered Boosting은 이를 방지하기 위해 Gradient boosting 알고리즘이 일부 수정되었다(Algorithm 10). Algorithm 10은 훈련데이터셋의 독립적인 $s+1$ 개의 무작위 순열을 생성하는데 이는 순열의 위치에 따라 최종 모델 예측의 분산을 증가시킬 수 있다. 이를 극복하기 위해 각 반복 t 의 $\{\sigma_1, \sigma_2, \sigma_3 \dots \sigma_s\}$ 에서 무작위 순열 σ_r 을 샘플링하여, 이것을 기반으로 트리 T 를 구축한다. σ_r 에 따라 모든 데이터로부터 TS와 잔차를 계산하고, 이를 독립적으로 유지하기 위해 $M_{r,j}$ 를 만든다. $M_{r,j}$ 는 순열 σ_r 에서 j 개의 인스턴스를 이용한 현재 모델을 말하며, $M_{r,j}(i)$ 는 σ_r 의 j 번째까지 데이터로 학습된 모델의 i 번째 데이터에 대한 예측값이고 그때의 기울기(gradient)는 코사인 유사도(cosine similarity)를 이용

하여 구하고 그 수식은 식 (2.35)과 같다. 잎 노드의 값(leaf value)인 $\Delta(i)$ 에 대해 이전 스텝의 트리 잎(leaf)의 잔차 평균을 할당하고 이전에 계산한 잔차와 새로 할당된 잔차 사이의 코사인 유사도가 적은 구간을 분할 구간(split point)으로 선정한다. 이 일련의 과정들을 반복하여 최종적으로 $M_{r,j}(i)$ 를 구축한다(Prokhorenkova et al., 2017; DSBA, 2020c).

$$grad_{r,j}(i) = \frac{\partial L(y_i, s)}{\partial s} \Big|_{s = M_{r,j}(i)} \quad (2.35)$$

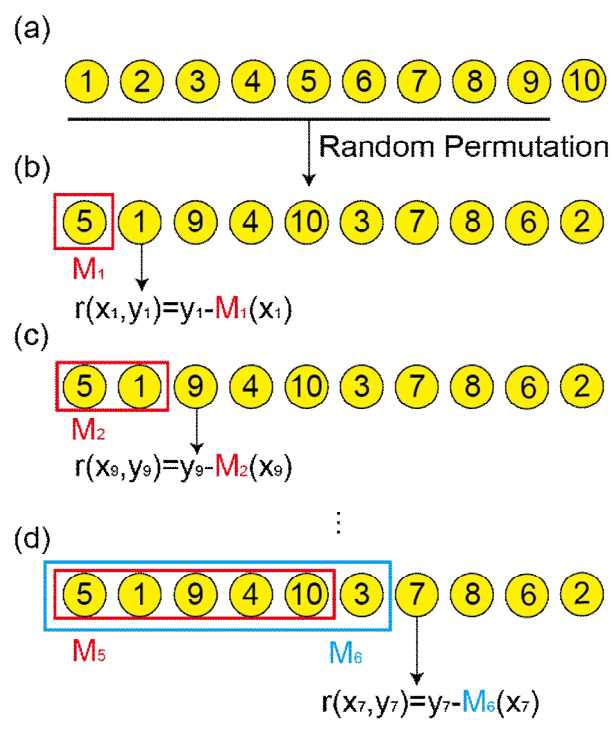


Figure 2-19. Ordered boosting according to random permutation (a) dataset before random permutation, (b) 1st step of calculating gradient, (c) 2nd step of calculating gradient, (d) 7th step of calculating gradient.

Algorithm 9. Ordered Boosting

```
input:  $(x_k, y_k)_{k=1}^n, I$ ;  
 $\sigma \leftarrow$  random permutation of  $[1, n]$ ;  
 $M_i \leftarrow 0$  for  $i=1 \dots n$ ;  
for  $t \leftarrow 1$  to  $I$  do  
  for  $i \leftarrow 1$  to  $n$  do  
     $\gamma_i \leftarrow y_i - M_{\sigma(i)-1}(x_i)$ ;  
  for  $i \leftarrow 1$  to  $n$  do  
     $\Delta M \leftarrow$   
    LearnModel( $(x_j, \gamma_j)$ ):  
     $\sigma(j) \leq i$ );  
     $M_i \leftarrow M_i + \Delta M$ ;  
return  $M$ 
```

Algorithm 10. Building a tree in CatBoost

```

input :  $M, (x, y), a, L, \{\sigma_i\}_{i=1}^s, Mode$ 
 $grad \leftarrow CalcGradient(L, M, y);$ 
 $r \leftarrow random(1, s);$ 
if  $Mode = Plain$  then
     $G \leftarrow (grad_r(i) \text{ for } i = 1 \dots n);$ 
if  $Mode = Ordered$  then
     $G \leftarrow (grad_{r, \sigma_r(i)-1}(i) \text{ for } i = 1 \dots n);$ 
 $T \leftarrow$  empty tree;
foreach step of top-down procedure do
    foreach candidate split  $c$  do
         $T_c \leftarrow$  add split  $c$  do
            if  $Mode = Plain$  then
                 $\Delta(i) \leftarrow avg(grad_r(p) \text{ for } p: leaf_r(p) = leaf_r(i)) \text{ for } i = 1 \dots n;$ 
            if  $Mode = Ordered$  then
                 $\Delta(i) \leftarrow avg(grad_{r, \sigma_r(i)-1}(p) \text{ for } p: leaf_r(p) = leaf_r(i), \sigma_r(p) < \sigma_r(i)) \text{ for } i = 1 \dots n;$ 
             $loss(T_c) \leftarrow cos(\Delta, G)$ 
         $T \leftarrow arg \min_{T_c} loss(T_c)$ 
    if  $Mode = Plain$  then
         $M_{r'}(i) \leftarrow M_{r'}(i) - \alpha avg(grad_{r'}(p) \text{ for } p: leaf_{r'}(p) = leaf_{r'}(i)) \text{ for } \gamma' = 1 \dots s, i = 1 \dots n;$ 
    if  $Mode = Ordered$  then
         $M_{r'}(i) \leftarrow M_{r'}(i) - \alpha avg(grad_{r', j}(p) \text{ for } p: leaf_{r'}(p) = leaf_{r'}(i), \sigma_{r'}(p) \leq j) \text{ for } \gamma' = 1 \dots s,$ 
         $i = 1 \dots n, j \geq \sigma_{r'}(i) - 1;$ 
return  $T, M$ 

```

2.4.2 CatBoost 학습 파라미터

CatBoost는 다른 GBM 기반 알고리즘과 데이터 훈련방식과 인코딩하는 방식 등에 의해 학습시간, 결측치 유무에 따른 훈련시간 소요, 그리고 정확도 등에서 큰 약점을 보인다. 이를 극복하기 위해 학습속도를 향상시키는 다양한 파라미터가 있다(Table 2-3).

Table 2-3. Parameters mainly used in CatBoost

Parameter	Description
boosting_type	Boosting scheme.
depth	Depth of tree.
rsm	Random subspace method. The percentage of features to use at each split selection, when features are selected over again at random.
min_data_in_leaf	The minimum number of training samples in a leaf.
iterations	The maximum number of trees that can be built when solving machine learning problems.
subsample	sample rate for bagging.
learning_rate	Used to reducing the gradient step. The learning rate.
l2_leaf_reg	Coefficient at the L2 regularization term of the cost function.
bagging_temperature	Defines the settings of the Bayesian bootstrap.
border_count	The number of splits for numerical features.
scale_pos_weight	The weight for class in binary classification. the value is used as a multiplier for the weights of objects from class 1.
sampling_frequency	Frequency to sample weights and objects when building trees.
objective	The metric to use in training.
eval_metric	The metric used for overfitting detection and best model selection.

3. 데이터 분석

3.1 비탈면 조사자료

비탈면 데이터는 국토 주변의 모든 깎기비탈면에 대한 데이터베이스를 산림청이나 한국건설기술연구원과 같은 관련 기관에 의해 구축되고 있으며, 그 자료는 매년 갱신되고 있다. 이렇게 구축된 데이터를 바탕으로 위험성이 높은 비탈면에 대해 전문가의 정밀조사가 수행되며, 그 결과로부터 적정대책 공법이 제안된다. 비탈면 데이터는 크게 두 가지로 분류된다. 첫 번째는 기초조사자료이며, 두 번째는 이를 바탕으로 수행되는 정밀조사에 의해 수집된 자료이다. 기초조사는 국토의 확장, 신설 노선 발생 등에 따른 새로운 비탈면이 준공되었을 때, 국토관리사무소(또는 지자체)로부터 행정구역, 구간, 연장 등 기초사항 정보를 받고, 비탈면 기초조사 매뉴얼에 따라 비탈면 기본정보를 직접 현장에서 취득하는 것을 말한다. 이러한 기초조사 자료는 위험한 깎기비탈면을 파악할 수 있는 기본 정보를 내포하고 있으며, 정밀조사 우선순위를 결정하는 주요한 자료가 된다. 반면 기초조사 이후에 수행되는 정밀조사는 비탈면의 불연속면 특성, 풍화도, 암반강도, 누수 현황, 배수시설 상태 등을 상세하게 조사하는 것으로 안정성 해석을 위한 기본 자료 및 대책공법 수립 등에 사용된다. 하지만 기초조사와 달리 정밀조사는 전문가에 의해 수행되기 때문에 전문가의 판단이 매우 중요하다. 이 과정에서 조사 수행자의 전문성 부족 및 자의적 해석 등으로 인해 잘못된 사례가 발생할 수 있다(MOLIT, 2018; Woo et al., 2020a; Woo et al., 2020b).

본 연구에서 사용한 비탈면 자료는 2006년부터 2020년까지 수집한 비탈면 자료로 샘플 수는 30,746개, 조사항목은 84개이다(Table 3-1). 조사항목에 대한 사항은 다음과 같다.

- 비탈면 관리코드 : 해당 비탈면을 식별할 수 있는 기초조사 코드.
- 관리청 및 관리기관 : 대상 비탈면을 관리하는 국토 유지건설사무소 및 지자체 도로사업소의 명칭.
- 도로호선 : 비탈면 주변 도로의 수.

- 행정구역 : 비탈면이 행정상 위치한 지역.
- 거리표 : 도로 관리를 위한 국도변 1 km 마다 설치된 거리.
- 위도, 경도 : GPS (Global Position System)를 활용한 비탈면의 좌표.
- 차선 : 비탈면이 위치한 도로의 차선.
- 조사일자 : 비탈면 현장 조사 일자.
- 사면길이 : 비탈면 밑단부터 끝단까지의 직선거리.
- 사면높이 : 비탈면 밑단부터 끝단까지의 수직거리.
- 사면경사 : 비탈면 밑단의 수평선이 인공으로 형성된(깎기) 비탈면 최정점과 이루는 각도.
- 사면상부경사 : 비탈면 상부 자연비탈면과 인공비탈면 최정점의 수평선이 이루는 각도.
- 이격거리 : 비탈면 끝단에서 길어깨까지의 거리.
- 소단 개소 : 비탈면 안정 및 유지관리를 위하여 높이에 따라 좁은 폭으로 조성되는 수평공간.
- 사면종류 : 비탈면의 구성 물질의 종류 또는 사면 높이와 토층심도의 비에 따라 암반비탈면, 토사비탈면, 혼합비탈면으로 분류.
- 주변지형 : 비탈면 주변 수계 및 산계의 특성을 산악, 준산악, 구릉, 평지로 구분.
- 지하수 : 비탈면에서 발생하는 누수의 정도와 위치.
- 풍화도 : 비탈면을 구성하는 암반의 풍화 등급.
- 불연속면 방향성 : 비탈면 내의 틈을 이루고 있는 절리면이 놓인 방향 또는 암석의 결 방향을 비탈면 경사방향과 일치, 수평방향, 비탈면 후방으로 향함, 확인할 수 없음 4가지로 구분.
- 사면형상 : 비탈면이 위치하는 도로의 종단현상 또는 비탈면방향으로 자른 단면의 형상. 블록형(철형), 직선형, 오목형(요형), 복합형으로 구분.
- 측면형상 : 비탈면을 측면에서 바라보았을 때의 형상. 비탈면을 세로 방향으로 절단하였다고 가정할 때 나타나는 형태. 돌출형, 요철형, 직선형, 탈락형으로 구분.
- 계곡부 개소 : 비탈면에 위치한 계곡부의 수.
- 붕괴 이력 : 과거 비탈면 내 붕괴, 낙석, 표층유실 등의 발생 흔적.
- 뜯돌 : 비탈면 내 암반이 느슨하게 매달려 있거나 후일에 빠져나오기 쉬운 바

- 위나 큰 돌. 크기나 양을 기준으로 무, 소, 중, 대로 구분.
- 낙석 : 암반 내 불연속면이 이완되어 암편이 모암으로부터 분리되어 낙하하는 현상. 낙석의 양, 낙석의 크기를 기준으로 무, 소, 중, 대로 구분.
 - 암종 : 비탈면의 주요 지질 및 암석.
 - 토층심도 : 토층과 기반암 사이의 평균 두께.
 - 암반형태 : 비탈면을 구성하는 암반의 크기와 모양에 의한 암반 분류. 비탈면의 리스크와 거동 특성 등을 파악할 수 있는 변수로서 절리가 거의 없거나 넓은 간격으로 존재하는 괴상(massive) 암반, 절리가 규칙적으로 존재해 암반이 거의 같은 크기로 존재하는 블록성(blocky) 암반, 절리가 심하게 발달된 파쇄(crushed) 암반, 암반의 크기가 다양한 불규칙(irregular) 암반, 절리의 한 면의 크기가 다른 두면 보다 적은 형태의 층상(tabular) 암반, 절리의 한 면의 크기가 다른 두 면보다 큰 형태의 층상(columnar) 암반으로 구분.
 - 불연속면 종류 : 비탈면에 발달되어 있는 여러 가지 형태나 크기의 약면. 암반이 연속체가 아닌 불연속체로 거동하도록 만드는 연약면의 총칭. 균열, 절리, 단층, 층리, 엽리, 편리, 파쇄대 등으로 구분.
 - 시공 현황/비탈면 보강공법 : 비탈면에 시공된 보강공법 및 보호공법의 종류와 시공된 그 공법의 현상태로 조사구간에 시공된 모든 공법에 대해 최대 7개까지 조사함.
 - 위험도 : 토석류 등 비탈면 파괴에 따른 위험 정도를 조사자 주관적 판단에 근거하여 ‘상’, ‘중’, ‘하’로 구분.
 - 피해도 : 토석류 등 비탈면 파괴에 따른 피해 정도를 조사자 주관적 판단에 근거하여 ‘상’, ‘중’, ‘하’로 구분.
 - 붕괴유형 : 비탈면 붕괴의 형태와 규모.
 - 위험구간 및 붕괴유형 : 비탈면의 위험구간의 위치와 규모 그리고 붕괴유형의 종류로 최대 3개의 위험구간에 대해 조사함.
 - 필요 주공법 : 조사자 판단에 근거하여 비탈면 파괴에 대한 대책안. 최대 3가지 방법을 제안함.
 - 조치 : 공법 적용의 필요성을 조사자 판단에 근거하여 ‘응급’과 ‘미응급’으로 구분.

Table 3-1. Slope investigation data for 15 years in Korea

	Investigation contents		Data type	Number of class(value)	Part
1	No.		Numerical	30747	Slope management and location information
2	Management code	Road number	Categorical	480	
3		Up/Down	Categorical	7	
4		Number	Numerical	4308	
5		Code name	Categorical	30742	
6	Administration		Categorical	5	
7	Management agency		Categorical	26	
8	Road line		Numerical	51	
9	Administration distriction	-do (province)	Categorical	27	
10		-si (city)	Categorical	163	
11		-dong, -eup (distriction)	Categorical	1020	
12		-ri (village)	Categorical	2506	
13	Distance mark	Start	Categorical	505	
14		End	Categorical	522	
15		km	Numerical	76	
16		m	Numerical	420	
17		Up/Down	Categorical	2	
18		Existence	Categorical	2	
19	Latitude	Degree	Numerical	7	
20		Minutes	Numerical	63	
21		Seconds	Numerical	1236	
22	Longitude	Degree	Numerical	7	
23		Minutes	Numerical	61	
24		Seconds	Numerical	1232	
25	Lane	Lane	Categorical	3	
26		Number of lane	Numerical	8	
27		Additional lane	Numerical	5	
28	Investigation date	Year	Numerical	15	
29		Month	Numerical	12	
30		Day	Numerical	31	

Table 3-1. Continued

	Investigation contents		Data type	Number of class(value)	Part
31	Slope length		Numerical	438	Slope characteristics
32	Slope height		Numerical	136	
33	Slope angle	Angle	Numerical	78	
34		Gradient	Numerical	57	
35	Angle of upper slope		Numerical	128	
36	Distance from road		Numerical	83	
37	Berm		Numerical	14	
38	Slope type		Categorical	6	
39	Topography		Categorical	4	
40	Groundwater		Categorical	9	
41	Leakage in vertical direction		Categorical	6	
42	Leakage in horizontal direction		Categorical	5	
43	Weathering		Categorical	7	
44	Direction of discontinuities		Categorical	6	
45	Slope shape		Categorical	4	
46	Shape of slope side		Categorical	4	
47	Valley		Numerical	8	
48	Collapse record		Categorical	5	
49	Floating rock		Categorical	4	
50	Rockfall		Categorical	4	
51	Lithology		Categorical	116	
52	Soil depth		Numerical	102	
53	Bedrock shape		Categorical	11	
54	Type of discontinuities		Categorical	13	
55	Stability method 1	Type	Categorical	94	
56		State	Categorical	6	
57	Stability method 2	Type	Categorical	132	
58		State	Categorical	7	
59	Stability method 3	Type	Categorical	166	
60		State	Categorical	6	
61	Stability method 4	Type	Categorical	142	
62		State	Categorical	6	
63	Stability method 5	Type	Categorical	110	
64		State	Categorical	4	
65	Stability method 6	Type	Categorical	81	
66		State	Categorical	5	
67	Stability method 7	Type	Categorical	17	
68		State	Categorical	1	

Table 3-1. Continued

	Investigation contents	Data type	Number of class(value)	Part	
69	Risk grade	Categorical	3	Investigator's opinion	
70	Damage grade	Categorical	5		
71	Slope stability	Categorical	7		
72	Danzer zone 3	Vertical direction	Categorical		8
73		Horizontal direction	Categorical		8
74		Type	Categorical		52
75	Danger zone 2	Vertical direction	Categorical		7
76		Horizontal direction	Categorical		6
77		Type	Categorical		22
78	Danger zone 3	Vertical direction	Categorical		3
79		Horizontal direction	Categorical		3
80		Type	Categorical		12
81	Required stability method	method 1	Categorical		211
82		method 2	Categorical		73
83		method 3	Categorical	17	
84	Follow-up	Categorical	4		

Table 3-1의 1번 ~ 30번은 비탈면의 행정구역, 관리번호, 위치 정보로서 비탈면 관리를 위한 정보이고, 31번 ~ 84번은 비탈면의 기초 및 정밀조사자료 결과에 해당한다. 그중 31번 ~ 68번은 비탈면의 특성에 대한 조사자료로, 수치형 데이터 9개와 범주형 데이터 39개로 구성되었다. Table 3-1의 69번 ~ 84번은 비탈면의 현 상태에 대한 전문가의 판단형 자료로 ‘낙석’, ‘뜯돌’, ‘붕괴이력’과 함께 비탈면 붕괴 발생과 관련한 인자이다. 그중 Table 3-1의 81번 ~ 84번은 비탈면의 안정성 확보에 대한 향후 계획에 해당한다. 현장 상황에 따라 중복조사 된 자료들이 존재한다. 대표적으로 ‘비탈면 보강공법’, ‘위험지역’, 그리고 ‘필요주공법’ 이에 해당한다. 각 비탈면의 여건에 따라 ‘비탈면 보강공법’은 최대 7개, ‘위험지역’과 ‘필요주공법’은 최대 3개까지 조사 및 제안되었다. 또한 수치형 데이터 9개를 제외한 대부분의 범주형 데이터들은 전문가에 의해 조사되었기 때문에 개인 의견과 작성 방법에 따라 인간오차(human error)가 발생하였다. 그 결과로 ‘비탈면 보강공법 3번의 공법 종류’에 166개의 항목이 조사되었다. 이처럼 범위가 넓은 데이터들은 모델 학습 시 훈련데이터에 과적합 될 가능성이 있으므로, 데이터 특성에 맞게 데이터 전처리 작업을 적용해야 한다.

비탈면 안정성 예측 모델을 위한 입력데이터는 비탈면의 특성과 관련 없는 자료인 위치 정보와 관리번호 등을 나타내는 Table 3-1의 1번 ~ 30번과 향후 계획 등을 의미하는 81번 ~ 84번을 제외한, 31번 ~ 80번의 총 50개 자료를 선정하였다. 출력데이터는 50개의 입력데이터 중에서 현재 비탈면의 붕괴 상태를 파악할 수 있는 ‘붕괴유형’으로 설정하였다. 데이터 전처리로 범주형 데이터의 구간화(또는 단순화)와 인코딩 그리고 수치형 데이터의 이상치 제거와 스케일링 등을 실시하였다.

3.2 범주형 데이터의 변수 구간화 및 인코딩

Table 3-1의 현장 조사자료와 머신러닝 모델을 이용하여 예측 모델을 평가하기 위해 범주형 변수에 대한 데이터 전처리를 실시하였다.

3.2.1 변수 구간화

Figure 3-1는 조사항목 중 ‘암종’의 상위 10개의 값을 나타낸 그림이다. 이 항목은 총 116개의 값이 할당되어있는 변수이다(Table 3-2). 이렇게 다양한 값이 할당되어있는 변수를 그대로 모델에 학습시킬 경우, 과적합 문제가 발생할 수 있으므로 변수를 단순화시킬 필요성이 있다. 예를 들면, ‘암종’의 경우 암석명 보다 ‘화성암’, ‘퇴적암’, ‘변성암’, ‘기타’ 등과 같이 암석 성인에 따라 분류하는 방안으로 데이터의 범위를 줄일 수 있다(Figure 3-2(m)). ‘비탈면 보강공법의 공법 종류’의 경우에는 ‘없음’, ‘보강’, ‘보호’, ‘그 외’로 총 4개의 클래스로 구간화(binining)하였다(Figure 3-2(q), (s), (u), (w), (y), (aa), (ac)). 또한 출력데이터인 ‘붕괴유형’은 조사자료 시 인간 오차로 인해 비탈면의 파괴 형태를 7가지로 조사되었다. 이를 ‘안정’과 ‘불안정’으로 이진 분류(binary classification)가 가능하도록 구간화하였다(Figure 3-2(ag)). 데이터 셋에 대한 각 변수의 구간화 결과는 Figure 3-2과 같다.

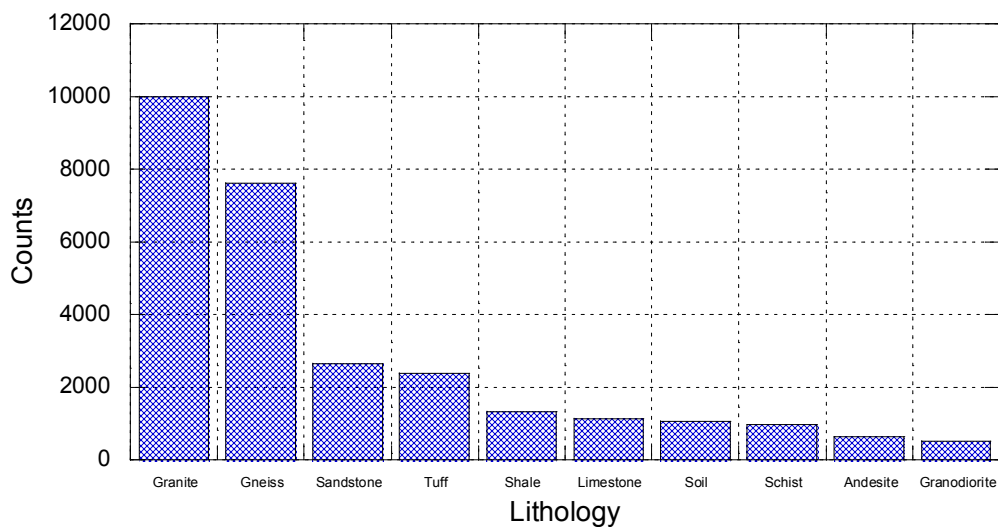


Figure 3-1. Top 10 values of lithology in the original dataset.

Table 3-2. The number of classes for each slope investigation contents

	Investigation contents		Data type	Number of class(value)	Part
1	Slope type		Categorical	3	Slope characteristics
2	Topography		Categorical	4	
3	Groundwater		Categorical	9	
4	Leakage in vertical direction		Categorical	4	
5	Leakage horizontal direction		Categorical	4	
6	Weathering		Categorical	6	
7	Direction of discontinuities		Categorical	4	
8	Slope shape		Categorical	4	
9	Shape of slope side		Categorical	4	
10	Collapse record		Categorical	2	
11	Floating rock		Categorical	4	
12	Rockfall		Categorical	4	
13	Lithology		Categorical	4	
14	Bedrock shape		Categorical	6	
15	Type of discontinuities		Categorical	5	
16	Stability method 1	Type	Categorical	4	
17		State	Categorical	5	
18	Stability method 2	Type	Categorical	4	
19		State	Categorical	5	
20	Stability method 3	Type	Categorical	4	
21		State	Categorical	5	
22	Stability method 4	Type	Categorical	4	
23		State	Categorical	5	
24	Stability method 5	Type	Categorical	4	
25		State	Categorical	5	
26	Stability method 6	Type	Categorical	4	
27		State	Categorical	4	
28	Stability method 7	Type	Categorical	4	
29		State	Categorical	2	
30	Risk grade		Categorical	3	Investigator's opinion
31	Damage grade		Categorical	3	
32	Slope stability		Categorical	2	

Table 3-2. Continued

	Investigation contents		Data type	Number of class(value)	Part
33	Danger zone 1	Vertical direction	Categorical	4	Investigator's opinion
34		Horizontal direction	Categorical	4	
35		Type	Categorical	5	
36	Danger zone 2	Vertical direction	Categorical	4	
37		Horizontal direction	Categorical	4	
38		Type	Categorical	5	
39	Danger zone 3	Vertical direction	Categorical	4	
40		Horizontal direction	Categorical	4	
41		Type	Categorical	5	

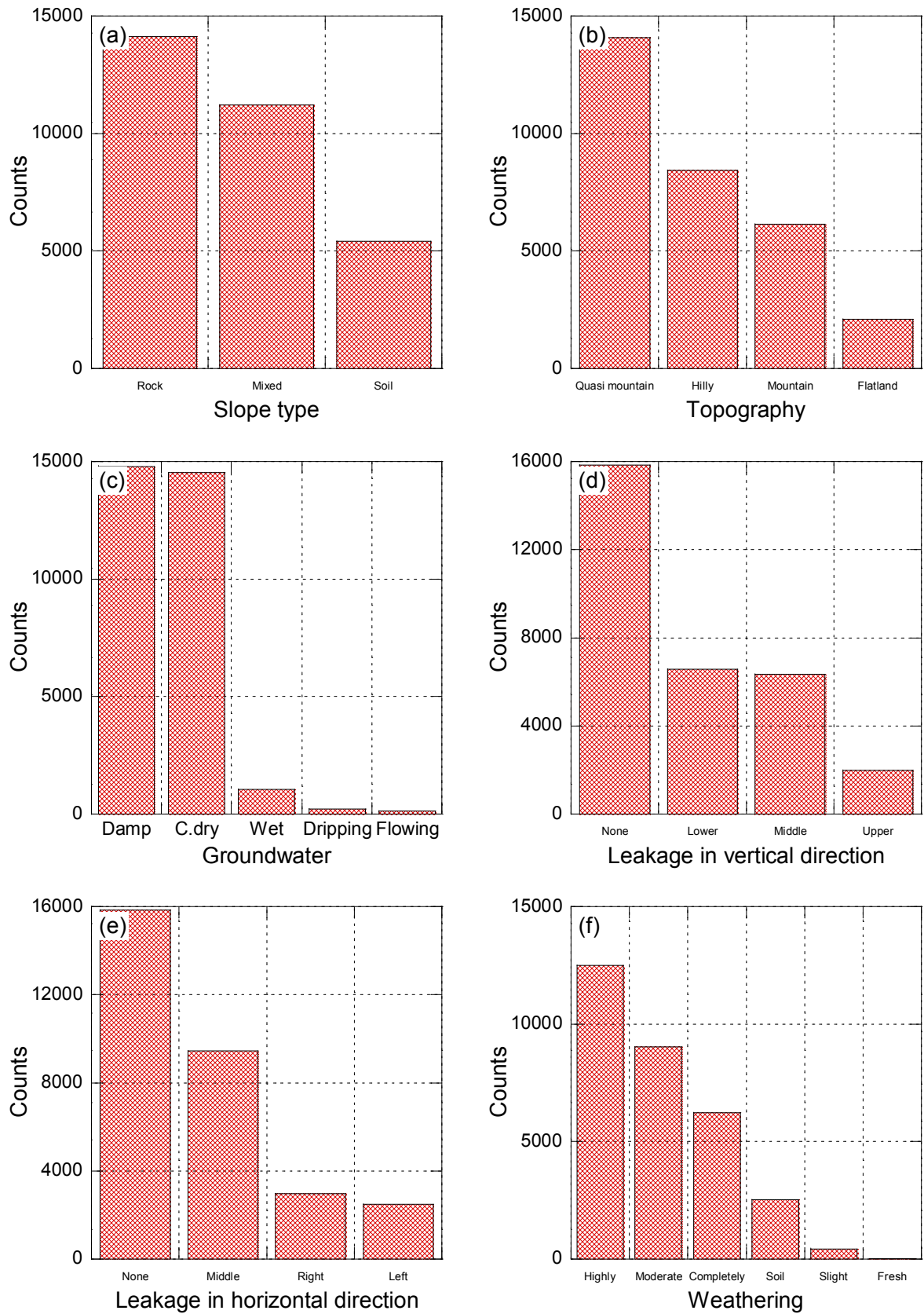


Figure 3-2. Distribution of simplified categorical values on each categorical data.

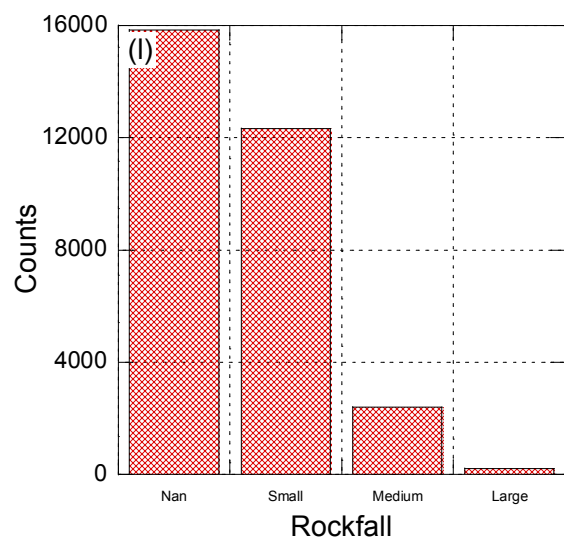
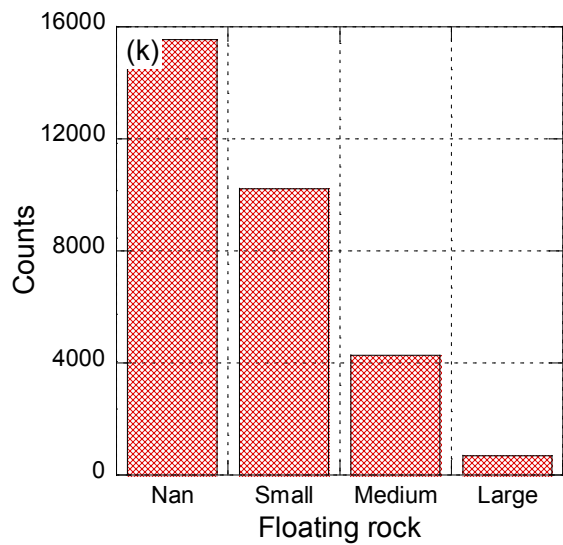
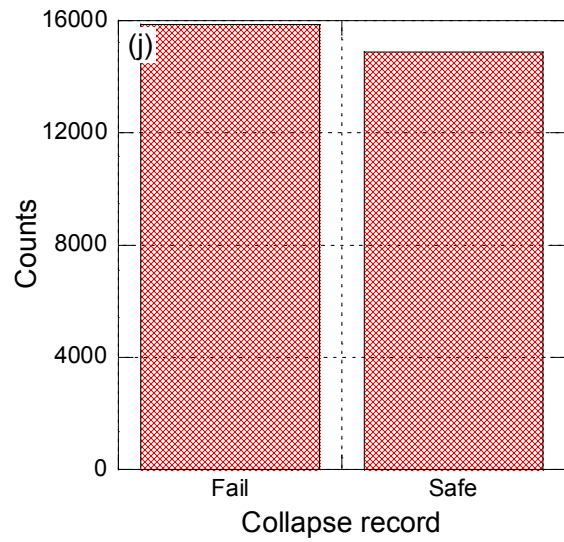
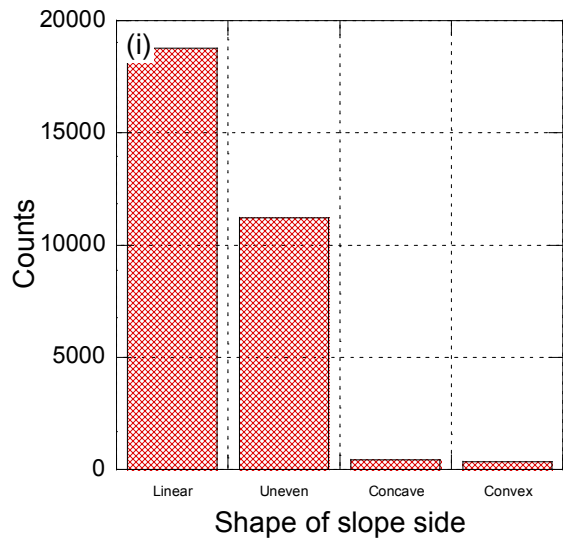
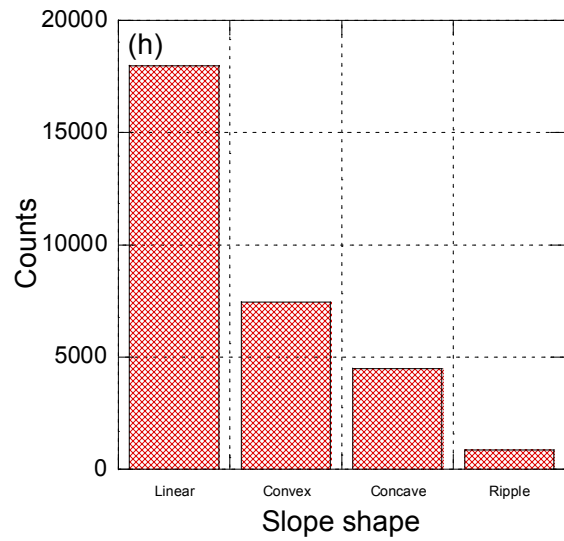
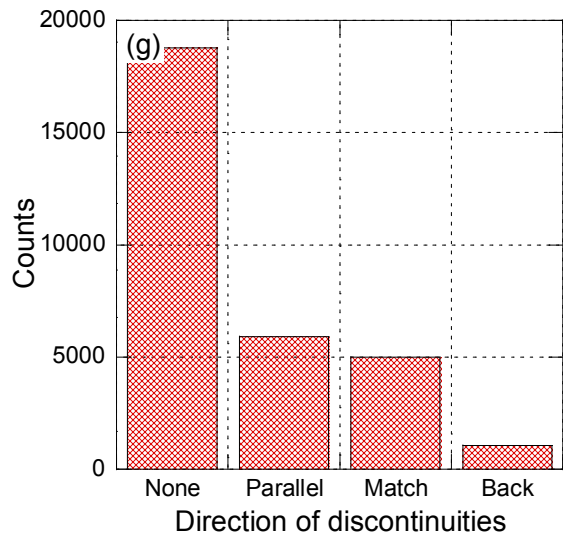


Figure 3-2. Continued.

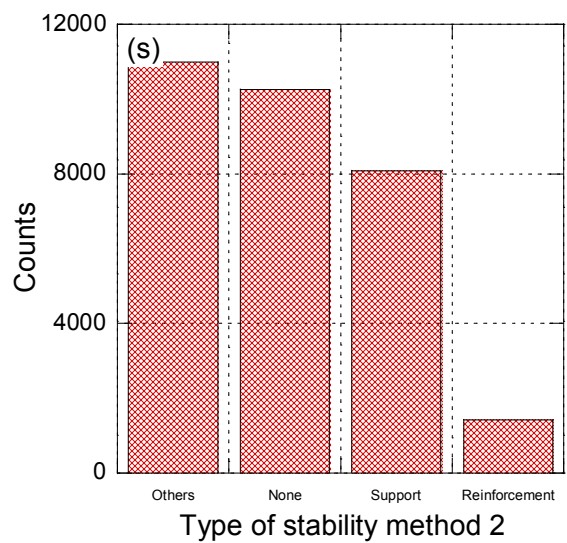
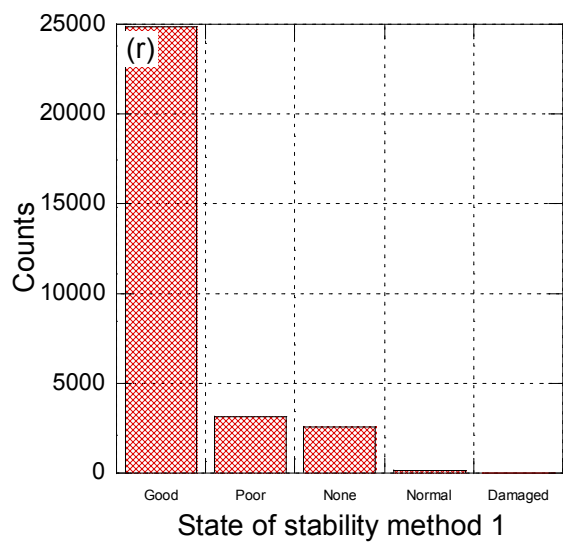
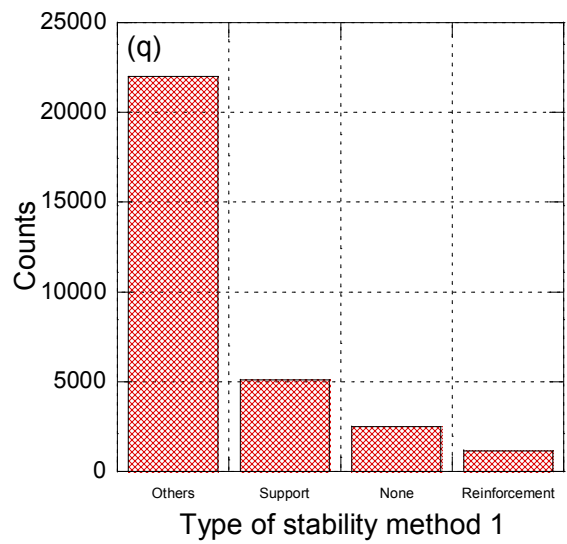
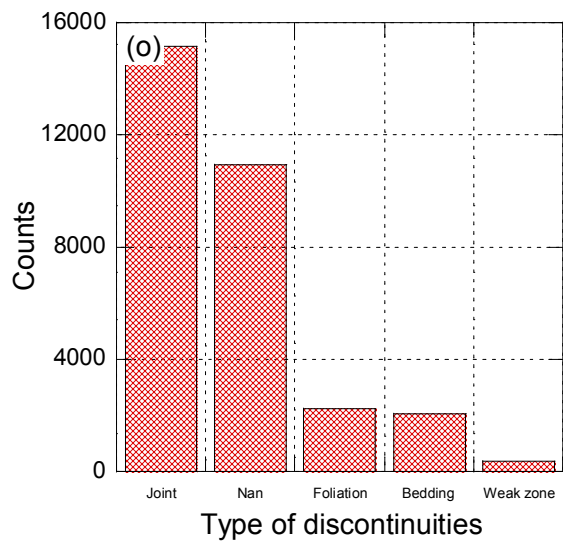
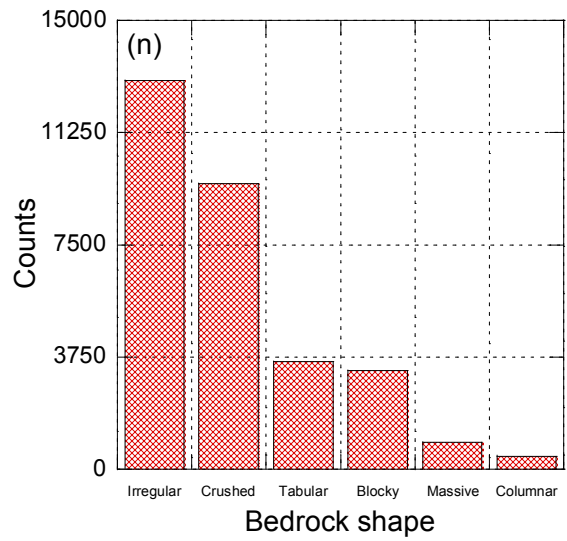
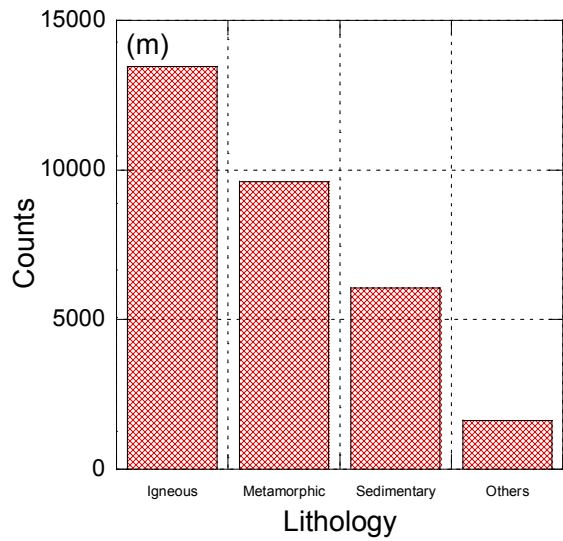


Figure 3-2. Continued.

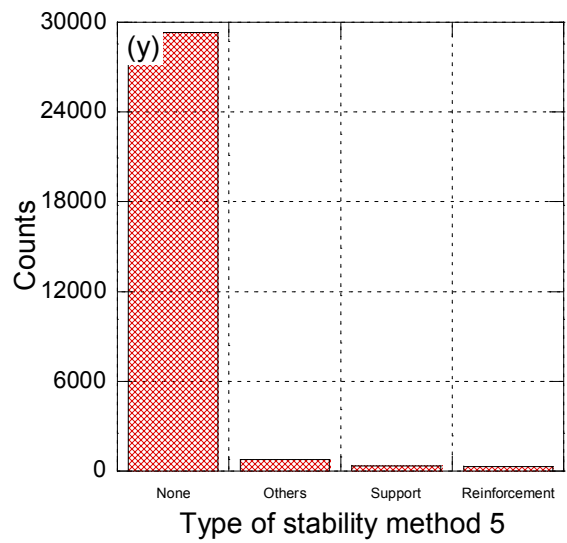
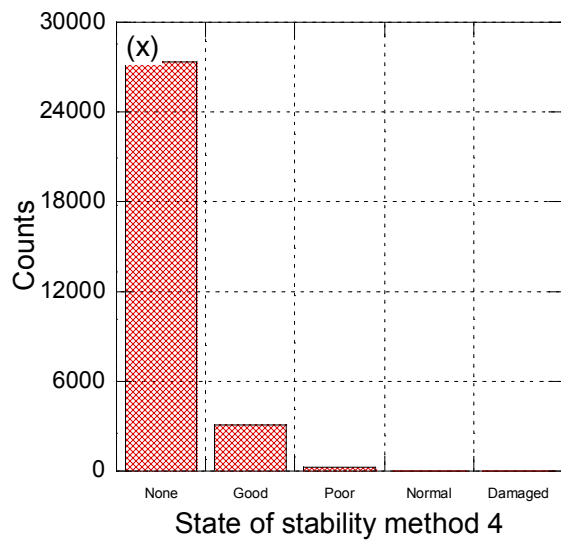
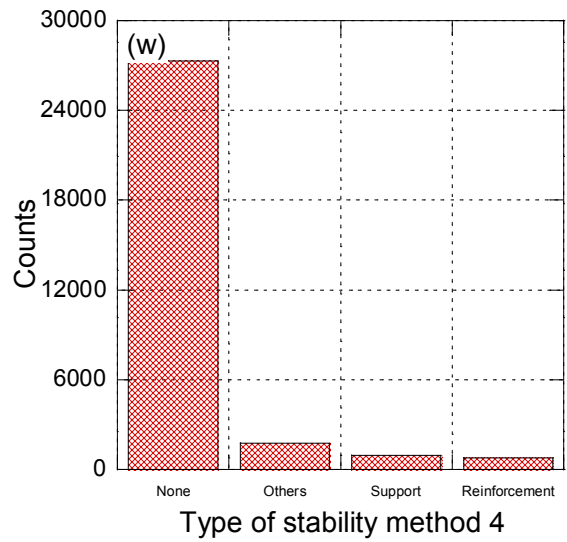
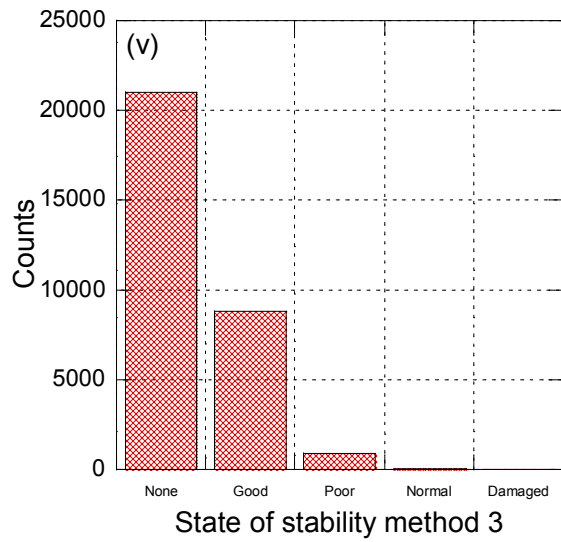
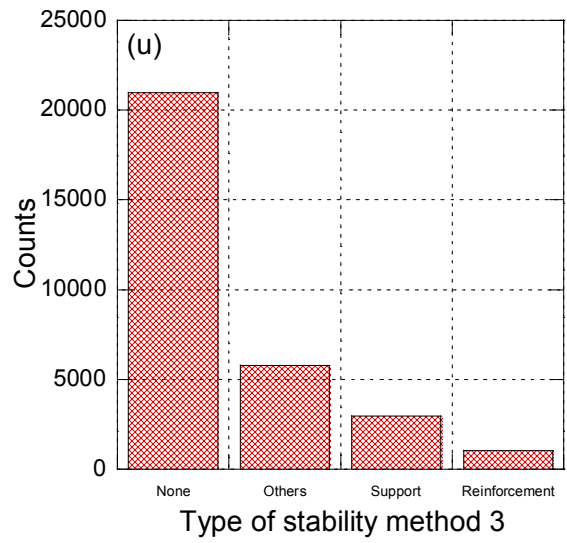
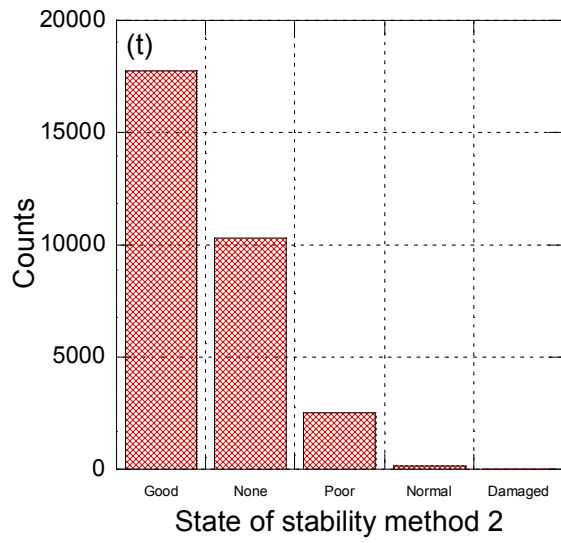


Figure 3-2. Continued.

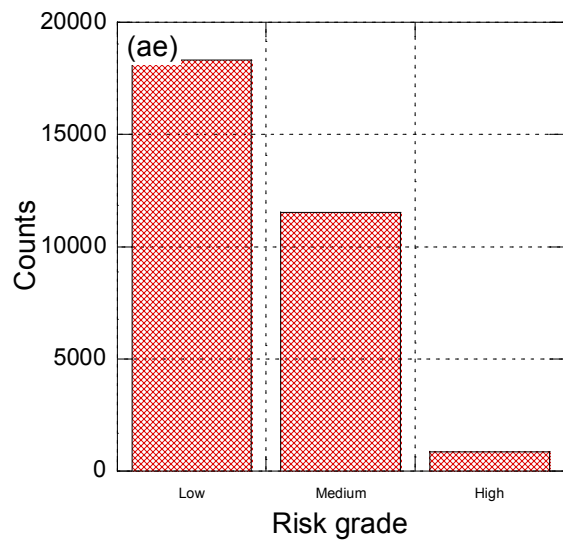
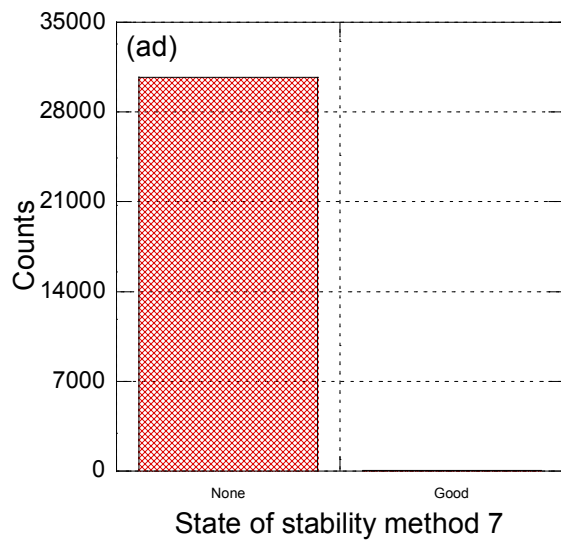
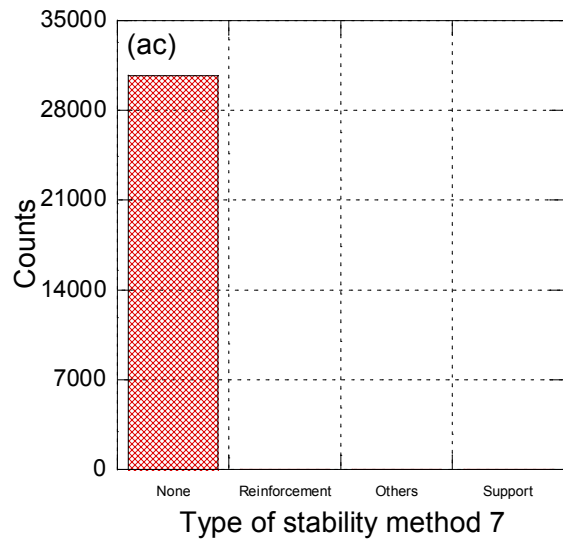
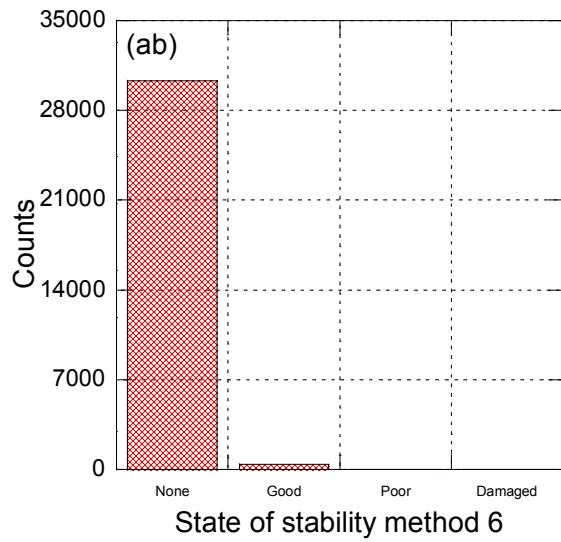
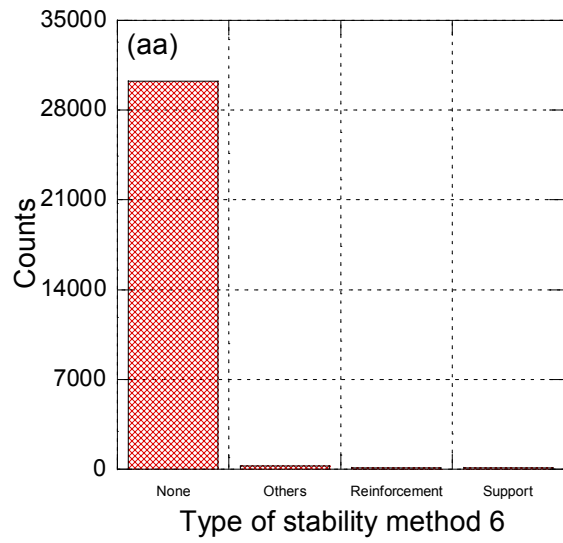
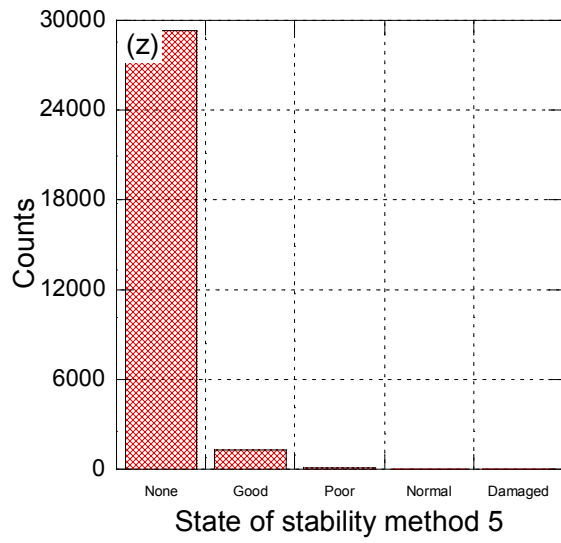


Figure 3-2. Continued.

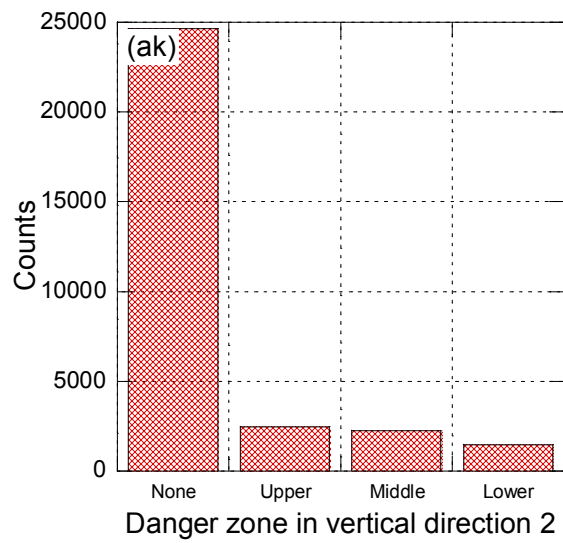
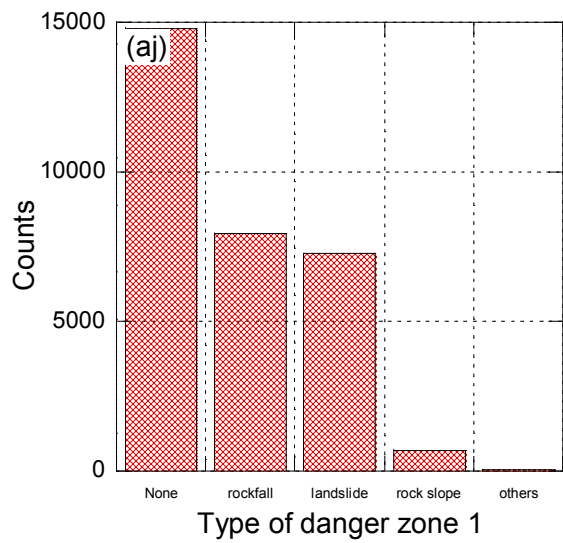
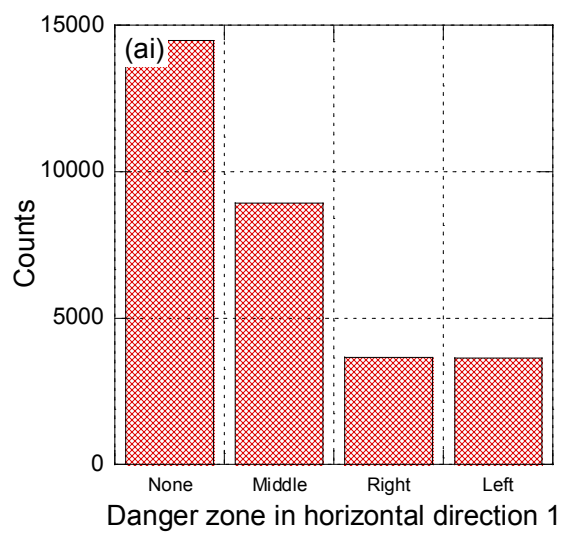
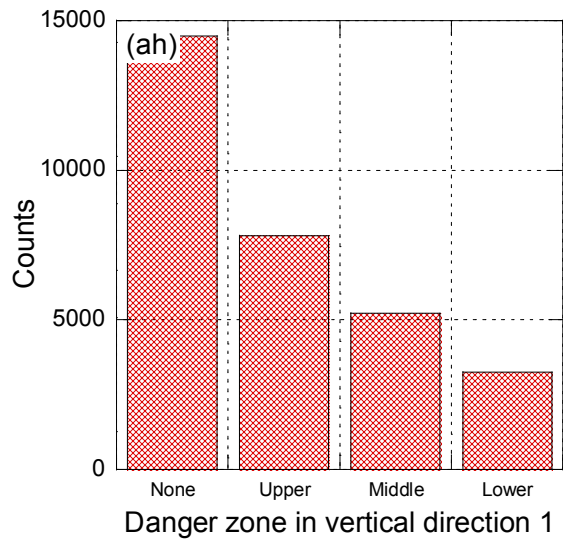
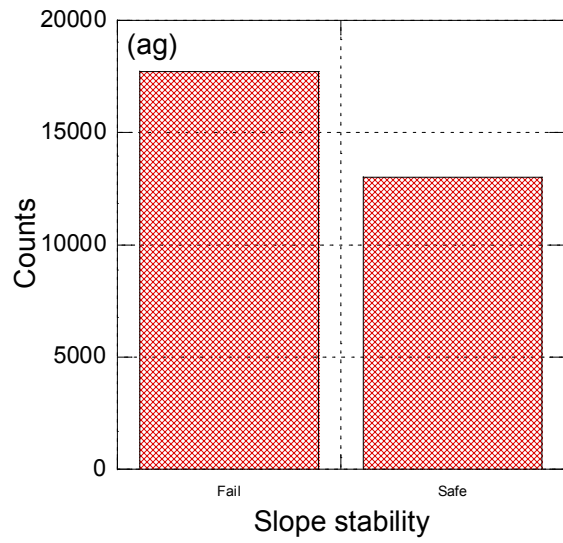
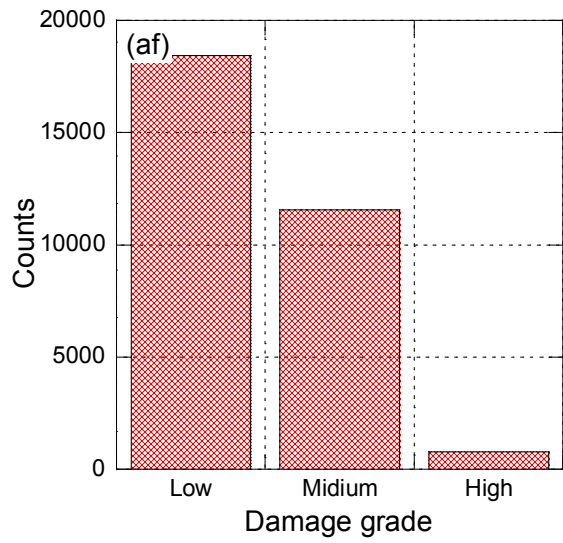


Figure 3-2. Continued.

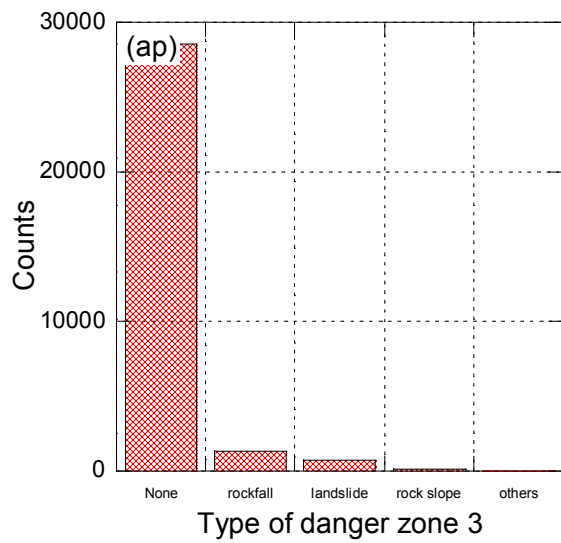
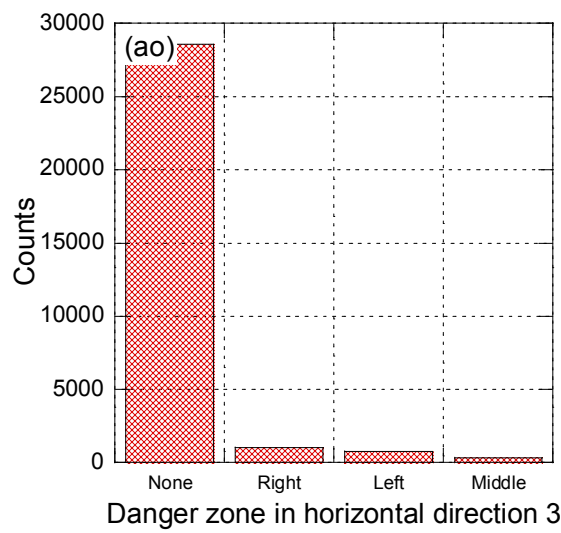
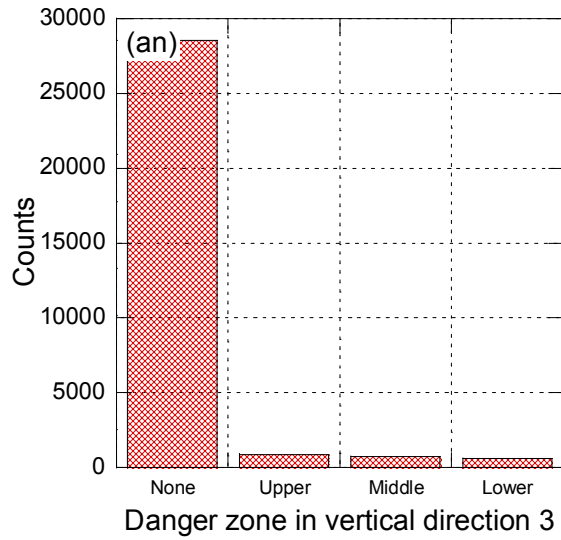
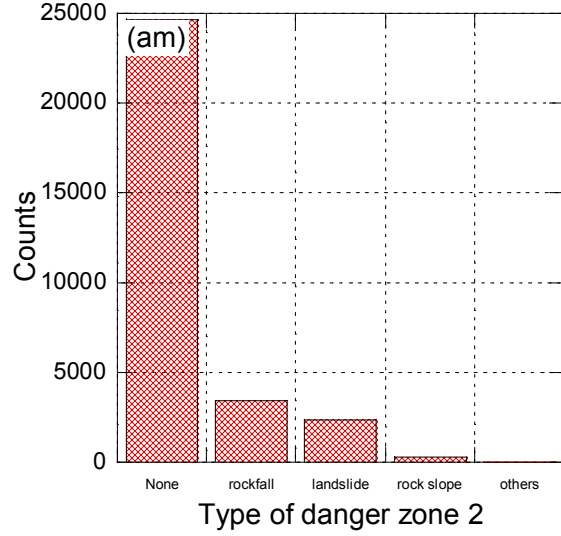
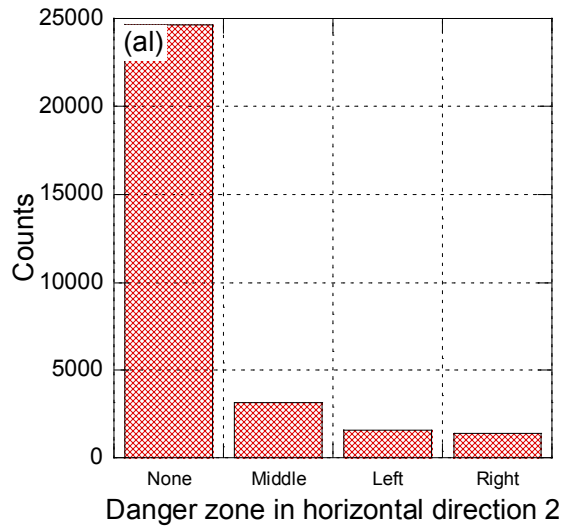


Figure 3-2. Continued.

3.2.2 변수 레이블링

머신러닝 모델들은 데이터 형태가 문자형일 경우 인식하지 못하기 때문에 문자를 숫자 형태로 인코딩(encoding) 해야 한다. 문자 형태로 이루어진 범주형 데이터는 개체들의 관계에 따라 우위가 존재하는 형식인 순서형(ordinal)과 우위가 존재하지 않는 명목형(nominal) 2가지로 나눌 수 있다. 인코딩할 때는 각 범주형 변수의 특성을 고려하여 적용해야 한다.

대표적인 인코딩 방법으로는 라벨 인코딩(label encoding)과 원-핫 인코딩(one-hot encoding)이 있다. 라벨 인코딩은 변수의 값(value or class)을 알파벳 순서에 따라 정수가 할당되는 방법이다. 데이터 자체의 속성을 숫자 형태로 바꾸기 때문에 데이터 크기와 모양은 바뀌지 않는다. 원-핫 인코딩은 변수의 크기를 벡터의 차원으로 가정하고, 해당하는 속성에 '1', 다른 인덱스에는 '0'을 부여하는 벡터 표현 방식이다. 이 방법은 속성의 크기가 클수록 벡터를 저장하기 위한 공간이 계속 늘어나는 단점이 있다. 인코딩 방식에 따라 모델이 알고리즘을 받아들이는 기준이 다르다. 인간은 해당 변수의 정보를 보고 파악이 가능하지만, 컴퓨터는 그렇지 못하기 때문이다. 이로 인하여 발생하는 정보손실을 최소화하기 위해 데이터 특징에 맞는 인코딩 방법을 적용함으로써 학습과 예측에서 좋은 성능을 보일 수 있다. 트리 계열 알고리즘도 숫자로 된 데이터 형태가 입력되면 그 데이터들을 숫자로 인식한다. 예를 들면, 개나 고양이를 인코딩하여 각각 1, 2로 변환시킨 뒤 학습시키면, 트리 계열의 알고리즘은 개보다 고양이가 큰 개체라고 인식하게 된다. 따라서 트리 계열 알고리즘에서 순서가 없는 명목형 데이터를 라벨 인코딩하고 학습시킨다면 성능이 잘 나오더라도 정보를 제대로 인식해서 학습하였다고 볼 수 없다. 따라서 해당 범주형 변수가 순서형인지 명목형인지를 고려해야 한다. 또 트리 계열의 알고리즘은 변수의 정보 이득이 최대가 될 때 가치를 분기하는데, 원-핫 인코딩의 경우 중요 변수가 변수값의 개수로 나뉘기 때문에 해당 변수가 높은 정보 이득을 가질 가능성은 원-핫 벡터로 표현된 값의 수만큼으로 분산되게 된다. 결국 알고리즘에 의해 잘 선택되지 않고 중요도도 낮아지게 된다. 변수 중요도는 모델 예측을 위한 절대적인 변수값은 아니지만, 해당 모델을 평가할 때 중요한 요소가 된다. 이러한 점을 고려하여 본 연구에서는 비탈면 안정성에 영향을 미치는 변수를 각 모델마다 비교 분석하기 위해 라벨 인코딩을 이용하여 범주형 데이터를 변환하였다.

3.3 수치형 데이터의 이상치 제거 및 스케일링

3.3.1 이상치 제거

이상치(Outlier)는 관측된 데이터의 전체적인 패턴이나 범위에서 벗어난 값을 의미하며, 모델의 예측 성능에 영향을 미칠 수 있으므로 제거할 필요성이 있다. 각 변수의 이상치를 제거해주기 위해 표준화(Standardization)를 진행하였다. 표준화를 적용하면 해당 데이터가 평균으로부터 얼마나 떨어져 있는지 파악할 수 있다. 표준화는 식 (3.1)로 구할 수 있으며, 각 변수의 평균을 0, 분산을 1로 변경하여 모든 피처가 같은 크기를 가지게 해야 한다. 여기서, z 는 표준점수 또는 Z-점수(Z-score), μ 는 평균, σ 는 표준편차를 나타낸다. Figure 3-3은 수치형 데이터 9개에 표준화를 적용한 결과를 보여준다. 이때 제1사분위수와 제3사분위수로 분류된 값들을 이상치로 분류하여 제거하였다.

$$z = \frac{x - \mu}{\sigma} \tag{3.1}$$

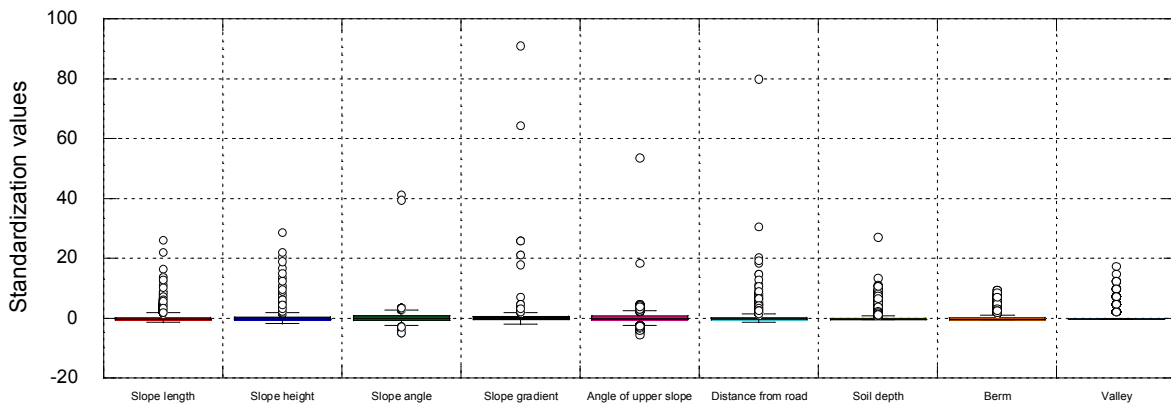


Figure 3-3. Distribution of each numerical data applied standardization.

Figure 3-3의 사면길이, 사면높이, 사면각도, 사면구배, 사면상부경사 그리고 토층심도의 이상치로 분류된 샘플을 제거해 주었지만, 소단과 계곡부 개소의 경우 이상치로 분류된 샘플의 수가 전체 데이터에서 차지하는 비율이 높아 제거하지 않았

다(Figure 3-4).

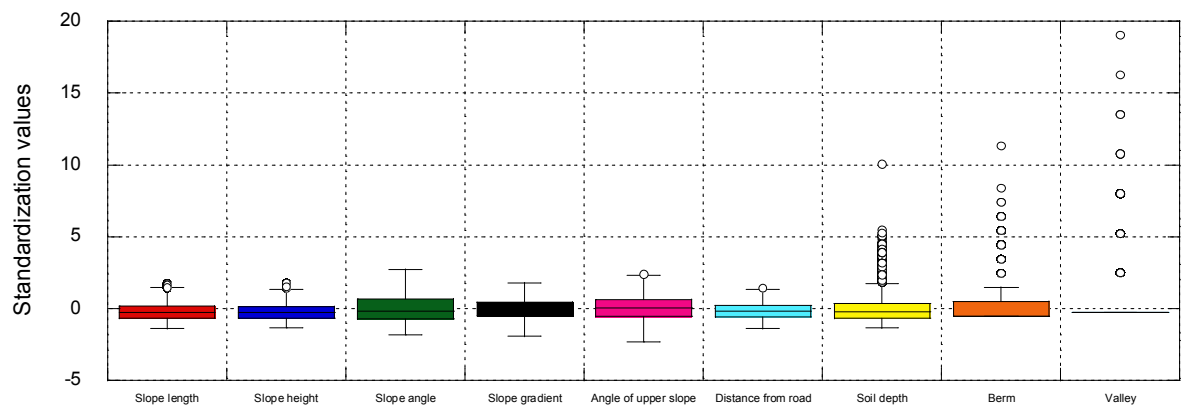


Figure 3-4. Distribution of standardized numerical data without outliers.

3.3.2 스케일링

데이터 스케일링(Data Scaling)은 각 변수들의 범위 혹은 분포를 같게 만드는 작업이다. 이를 통해 각 변수들이 동일한 조건을 가지게 되면 이 변수들에 대한 상대 비교가 가능하게 된다. 데이터 스케일링에는 표준화(Standardization), Min Max scaling, Max Abs scaling, Robust scaling 등이 있다.

(i) Min Max Scaling

식 (3.2)로 표현되는 Min Max Scaling은 각 변수 특성을 특정 범위(주로 0~1)로 스케일링하는 방법이다. 즉, 가장 작은 값은 '0', 가장 큰 값은 '1'로 변환된다. 하지만 이 방법은 이상치에 매우 민감하므로 주로 분류 보다 회귀에서 사용된다.

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \quad (3.2)$$

(ii) Max Abs Scaling

식 (3.3)으로 표현되는 Max Abs Scaling은 각 특성의 절대값이 0과 1 사이가 되도록 스케일링하는 방법이다. 모든 값이 -1~1의 값을 갖게 되며, 이 방법도 이상치에 매우 민감하다.

$$x' = 2 \left(\frac{x - x_{\min}}{x_{\max} - x_{\min}} - 0.5 \right) \quad (3.3)$$

(iii) Robust Scaling

식 (3.4)로 표현되는 Robust Scaling은 평균과 분산 대신 중간값과 사분위값을 사용하는 방법으로 이상치에 미치는 영향을 최소화하는 방법이다.

$$x' = \frac{(x - x_{median})}{x_{Q3} - x_{Q1}} \quad (3.4)$$

표준화를 사용해 이상치를 제거하는 경우 대부분 Min Max Scaling이 사용되는데, 그 이유는 모든 특성을 0과 1 사이의 동등한 기준으로 비교할 수 있기 때문이다. 하지만 본 연구에서는 소단과 계곡부 개소의 이상치로 분류된 샘플을 제거하지 못하였므로, Robust Scaling 방법을 적용하여 모델에서 발생하는 이상치의 영향을 최소화하였다(Figure 3-5).

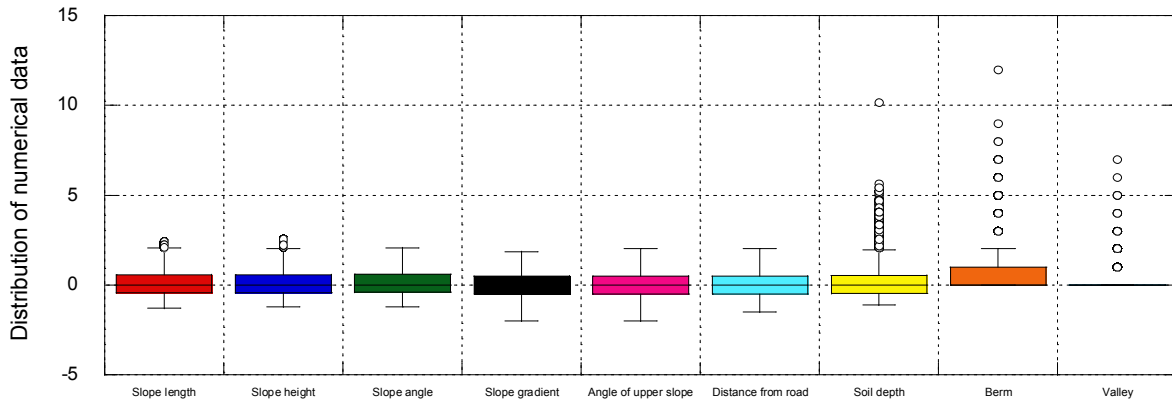


Figure 3-5. Distribution of scaled numerical data.

3.4 교차 분석

교차 분석(cross tabulation analysis)은 두 범주형 변수 간의 관계가 상호 독립 관계인지, 아니면 상호 연관성을 맺고 있는지를 검증하는 통계적 방법이다. 이 중 카이제곱(chi-square) 교차 분석이 가장 많이 사용되고 있다. 이 방법은 실제 관찰 빈도와 통계적으로 기대할 수 있는 빈도인 기대빈도 간의 차이를 카이제곱분포(chi-squared distribution)를 이용하여 통계적으로 검증하는 기법이다. 여기서 기대빈도는 모집단의 빈도 모수와 일치하는 값을 나타낸다. 만일 각 변수에 대한 관찰 빈도와 기대빈도가 거의 차이가 없다면 두 변수는 상호 독립으로 평가할 수 있고, 그 차이가 크면 독립성이 없고 연관성이 있다고 평가할 수 있다. 기대빈도와 카이제곱 통계량을 산출하는 공식은 식 (3.5)와 식 (3.6)으로 나타낼 수 있다. 여기서, E는 기대빈도로서, 데이터의 행과 열의 각 관측치의 합을 곱한 후 총빈도수로 나눈 것이다. C와 R은 각 행과 열에 속한 관측치의 수, N은 전체 관측치의 수, O는 관찰빈도를 나타낸다. 따라서 검증 통계량인 카이제곱은 관찰빈도인 O와 E의 차이를 곱한 후 기대빈도로 나눈 값을 합산한 값이다.

$$E = \frac{(C+R)}{N} \quad (3.5)$$

$$\chi^2 = \sum \frac{(O-E)^2}{E} \quad (3.6)$$

본 연구에서는 모든 범주형 형태의 입력데이터와 출력데이터인 ‘붕괴유형(slope stability)’과의 연관성을 알아보기 위해 교차 분석을 실시하였고, 유의 수준은 일반적으로 많이 사용하는 5%로 설정하였다. 교차 분석 결과 유의확률(p-value)이 5% 이상으로 나타난 변수는 ‘비탈면 보강공법 6번의 공법 종류’와 ‘비탈면 보강공법 7번의 공법 종류’, ‘비탈면 보강공법 6번의 상태’와 ‘비탈면 보강공법 7번의 상태’으로 나타났다(Table 3-3). 따라서, 4개의 입력데이터들은 ‘붕괴유형’과 관련이 없는 것으로 나타났고, 입력데이터 49개 중에서 이 4개의 변수를 제외한 총 45개를 입력데이터로 사용하였다.

Table 3-3. Results for cross tabulation of categorical data

Input variables		p-value of cross tabulation analysis
Slope type		0.00
Topography		0.00
Groundwater		0.00
Leakage in vertical direction		0.00
Leakage in horizontal direction		0.00
Weathering		0.00
Direction of discontinuities		0.00
Slope Shape		0.00
Side shape of slope		0.00
Collapse record		0.00
Floating rock		0.00
Rockfall		0.00
Lithology		0.00
Bedrock shape		0.00
Discontinuities		0.00
Stability method 1	Type	0.00
	State	0.00
Stability method 2	Type	0.00
	State	0.00
Stability method 3	Type	0.00
	State	0.00
Stability method 4	Type	0.00
	State	0.00
Stability method 5	Type	0.00
	State	0.00
Stability method 6	Type	0.06
	State	0.40
Stability method 7	Type	0.49
	State	0.25

Table 3-3. Continued

Input variables		p-value of cross tabulation analysis
rick grade		0.00
damage grade		0.00
Danger zone 1	Vertical direction	0.00
	Horizontal direction	0.00
	Type	0.00
Danger zone 2	Vertical direction	0.00
	Horizontal direction	0.00
	Type	0.00
Danger zone 3	Vertical direction	0.00
	Horizontal direction	0.00
	Type	0.00

3.5 상관관계 분석

각 입력변수들의 관계를 정량적으로 분석하기 위해 각 변수들 간의 상관관계를 분석하였다. 상관관계는 두 변수 간의 관계를 $-1 \sim 1$ 사이의 수치로 표현한 값이며, -1 이나 1 에 가까울수록 상관성이 높고, 0 에 가까울수록 상관성이 낮다. 각 변수들의 상관관계는 식 (3.7)의 피어슨(Pearson) 상관계수를 사용하였다.

$$r_{xy} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}} \quad (3.7)$$

여기서, n 은 샘플 데이터의 수, x_i 는 i 번째 x 값, y_i 는 i 번째 y 의 값, \bar{x} 는 변수 x 의 평균, \bar{y} 는 변수 y 의 평균을 나타낸다. 상관관계 분석 결과는 Figure 3-6과 같고, ‘붕괴유형’과 상관관계가 높은 변수는 비탈면 현장 조사자료 중 전문가의 판단형 자료들로서 비탈면 붕괴 발생과 관련한 변수들인 ‘붕괴이력’, ‘뜯돌’, ‘낙석’, ‘위험도’, ‘피해도’, ‘위험구간 1번의 세로 방향’, ‘위험구간 1번의 가로 방향’, ‘위험구간 1번의 종류’, ‘위험구간 2번의 세로 방향’, ‘위험구간 2번의 가로 방향’, ‘위험구간 2번의 종류’, ‘위험구간 3번의 세로 방향’, ‘위험구간 3번의 가로 방향’, ‘위험구간 3번의 종류’ 등의 총 14개로 나타났다.

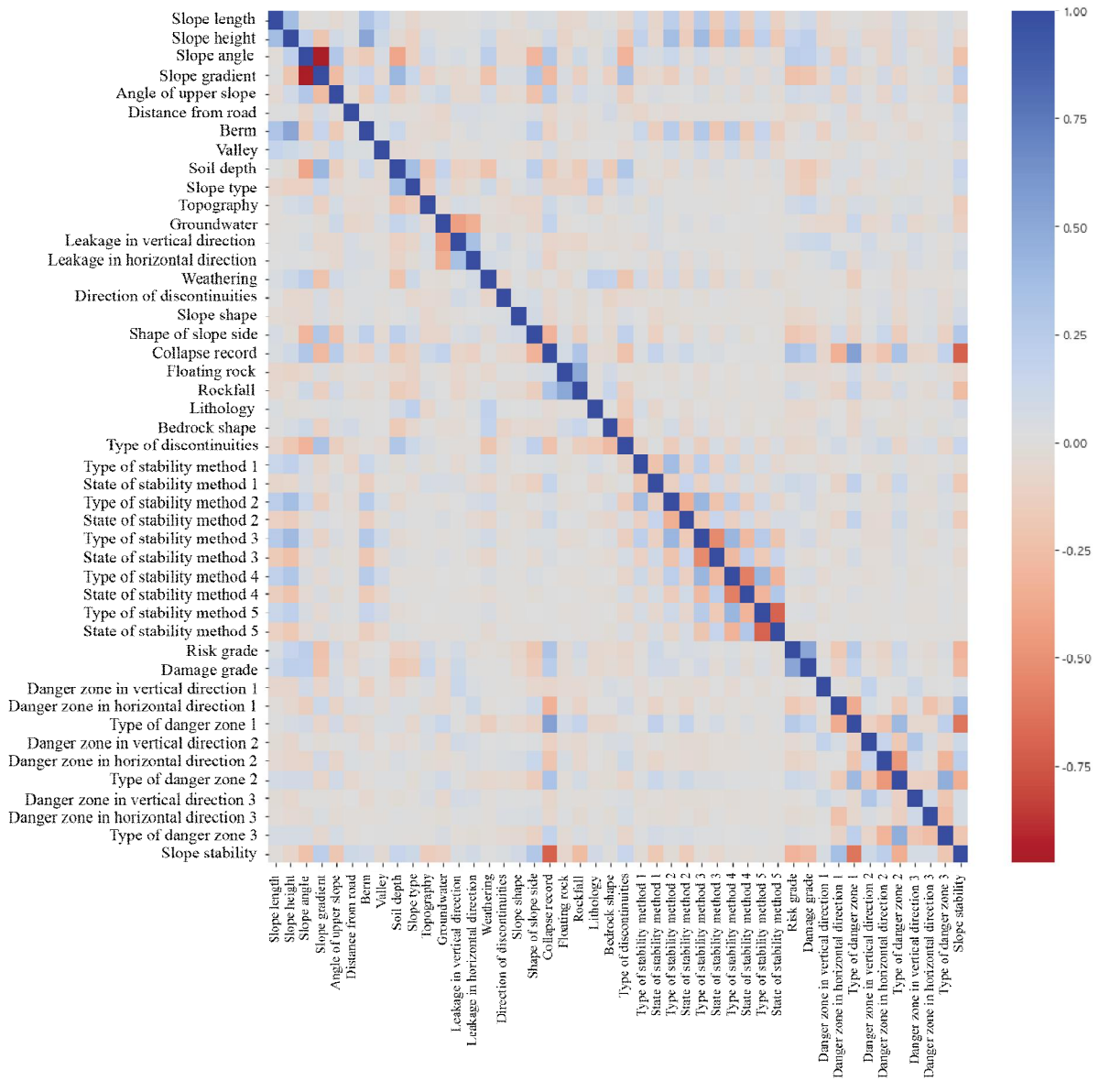


Figure 3-6. Correlation of variables in slope investigation data.

3.6 해석 단계 설정

분류에서는 일반적으로 상관계수가 높게 나타나면 모델의 예측 성능이 높아진다. 하지만 입력데이터가 출력데이터의 역할을 하거나, 공분산으로 인해 예측력을 저하시키는 등의 몇몇 경우에는 잡음으로 판정해 그 변수들을 제외하는 예도 있다. 즉, ‘위험도’, ‘피해도’, ‘붕괴 이력’ 등 14개의 상관관계가 높은 변수들이 모델의 예측 성능을 향상시켜도, 비탈면 안정성 예측 모델을 사용할 때 비탈면 특성 자료들의 비중이 줄어들면 안정성 예측 모델로서의 가치를 상실하게 된다. 본 실험에 앞서 입력데이터를 최종적으로 선정하고, 해석 단계를 설정하기 위해 로지스틱 회귀분석을 사용하여 비탈면의 안정성을 예측하였다. 로지스틱 회귀분석(Logistic Regression, LR)은 이진분류를 해결할 수 있는 가장 쉬운 도구 중 하나로 데이터가 어떤 범주에 속할 확률을 0~1 사이의 값을 예측하고 그 확률에 따라 가능성이 더 큰 범주에 속하는 것으로 분류해주는 지도학습 알고리즘이다. 로지스틱 회귀분석은 식 (3.8)과 같은 Sigmoid 함수를 이용하여 확률을 계산한다.

$$h(x) = \frac{1}{1 + e^{-x}} \quad (3.8)$$

해석 단계를 설정하기 위해 상관계수가 높은 변수 14개를 포함한 총 45개의 입력데이터를 이용한 LR-1과 14개의 변수를 포함하지 않은 총 31개의 입력데이터를 이용한 LR-2로 구분해 로지스틱 회귀분석을 이용해 분류 성능을 비교하였다(Table 3-4, 3-5). 이때 사용한 매개변수는 모두 기본값(default)을 이용하였고, 훈련과 시험데이터의 비율은 7:3으로 설정하였다. 모델의 예측 성능 평가는 정확도와 AUC를 이용하였다.

Table 3-4. Datasets of the LR-1 analysis model

		Input variables			Output variable
LR-1	Slope length	Slope height	Slope angle	Slope stability	
	Slope gradient	Angle of upper slope	Distance from road		
	Berm	Valley	Soil depth		
	Slope type	Topography	Groundwater		
	Leakage in vertical direction	Leakage in horizontal direction	Type of discontinuities		
	Direction of discontinuities	Lithology	Slope shape		
	Bedrock shape	Weathering	Side shape of slope		
	Type of stability methods 1	State of stability methods 1	Type of stability methods 2		
	State of stability methods 2	Type of stability methods 3	State of stability methods 3		
	Type of stability methods 4	State of stability methods 4	Type of stability methods 5		
	State of stability methods 5	Type of danger zone 1	Danger zone 1 in vertical direction		
	Danger zone 1 in horizontal direction	Type of danger zone 2	Danger zone 2 in vertical direction		
	Danger zone 2 in horizontal direction	Type of danger zone 3	Danger zone 3 in vertical direction		
	Danger zone 2 in horizontal direction	Rockfall	Floating rock		
	Risk grade	Damage grade	Collapse record		

Table 3-5. Datasets of the LR-2 analysis model

		Input data			Output data
LR-2	Slope length	Slope height	Slope angle	Slope stability	
	Slope gradient	Angle of upper slope	Distance from road		
	Berm	Valley	Soil depth		
	Slope type	Topography	Groundwater		
	Leakage in vertical direction	Leakage in horizontal direction	Type of discontinuities		
	Direction of discontinuities	Lithology	Slope shape		
	Bedrock shape	Weathering	Side shape of slope		
	Type of stability methods 1	State of stability methods 1	Type of stability methods 2		
	State of stability methods 2	Type of stability methods 3	State of stability methods 3		
	Type of stability methods 4	State of stability methods 4	Type of stability methods 5		
	State of stability methods 5				

비탈면 안정성 예측 모델의 입력데이터를 결정하기 위하여 해석 단계를 2단계로 구분해 로지스틱 회귀분석 한 결과는 Table 3-6과 Figure 3-7, 3-8과 같다. LR-1의 정확도는 89%, AUC는 0.946으로 나타났고, LR-2의 경우 정확도는 71%, AUC는 0.777로 나타났다. LR-1과 LR-2의 예측 성능 지표들의 결과 차이가 크게 발생한 것을 고려해 볼 때, 비탈면 붕괴와 관련한 14개의 변수들이 LR-1 모델에서 출력데이터와 유사한 역할을 하는 것으로 판단되며 또한 비탈면 안정성에 영향을 미치는 특성을 파악하는데 어려움이 있을 것으로 사료된다. 따라서 본 연구에서는 이들 변수를 입력변수에서 제외시켰다.

Table 3-6. Results for logistic regression according to the analysis steps

	Accuracy (%)	AUC
LR-1	89	0.946
LR-2	71	0.777

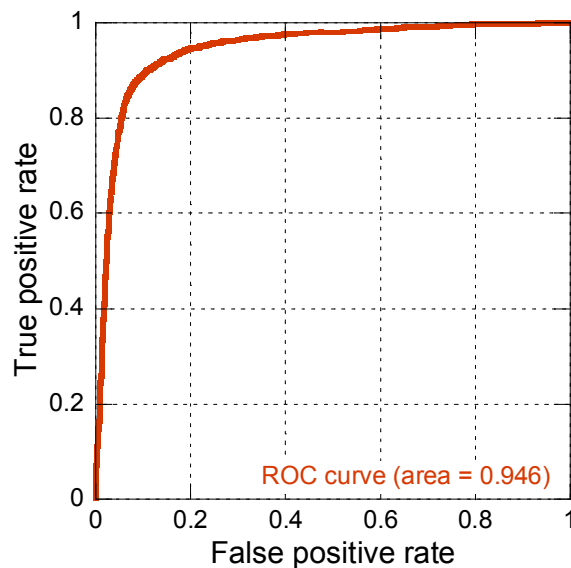


Figure 3-7. AUC of LR-1 as a result of logistic regression.

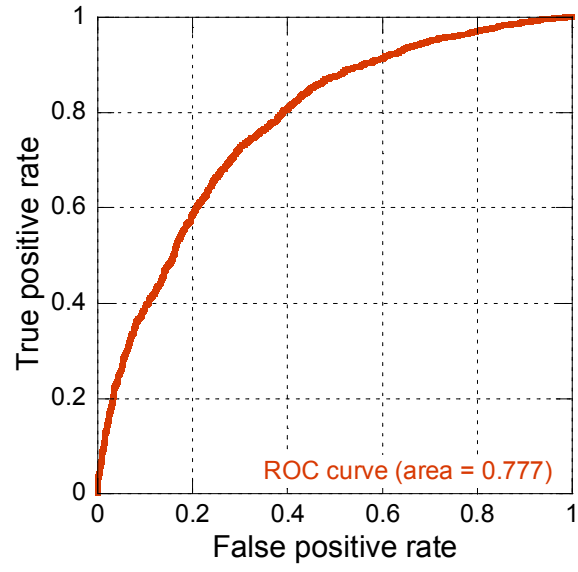


Figure 3-8. AUC of LR-2 as a result of logistic regression.

출력 변수인 ‘붕괴유형’은 Figure 3-9(a)와 같이 ‘불안정’은 ‘0’, ‘안정’은 ‘1’로 인코딩되었다. ‘0’은 15,092개로 ‘1’의 10,806개보다 4,286개 많다. 이러한 데이터 불균형은 예측 성능에 악영향을 미치기 때문에 균형을 맞추어야 한다. 따라서, 붕괴유형의 두 클래스를 10,806개로 동일하게 조절하였다(Figure 3-9(b)).

비탈면 안정성 예측 모델을 평가하기 위해 데이터 전처리를 실시하였고, 모델 학습을 위한 입력데이터와 출력데이터를 선정하였다. 선정된 입력데이터는 수치형 데이터 9개와 범주형 데이터 22개로 총 31개의 변수로 구성되었다. 수치형 데이터는 비탈면 길이, 높이 등 비탈면의 크기를 의미하는 객관적 자료에 해당하고, 범주형 데이터는 전문가의 현장 조사 결과로 비탈면의 질적 상태를 나타낸다. 각각의 머신러닝 모델과 비탈면 조사자료의 특성에 따른 변수의 중요도를 비교분석하기 위하여 해석 단계를 수치형 데이터 9개로 구성된 Case-1, 범주형 데이터 22개로 구성된 Case-2, 그리고 전체 데이터 31개로 구성된 Case-3로 구분하였다(Table 3-7).

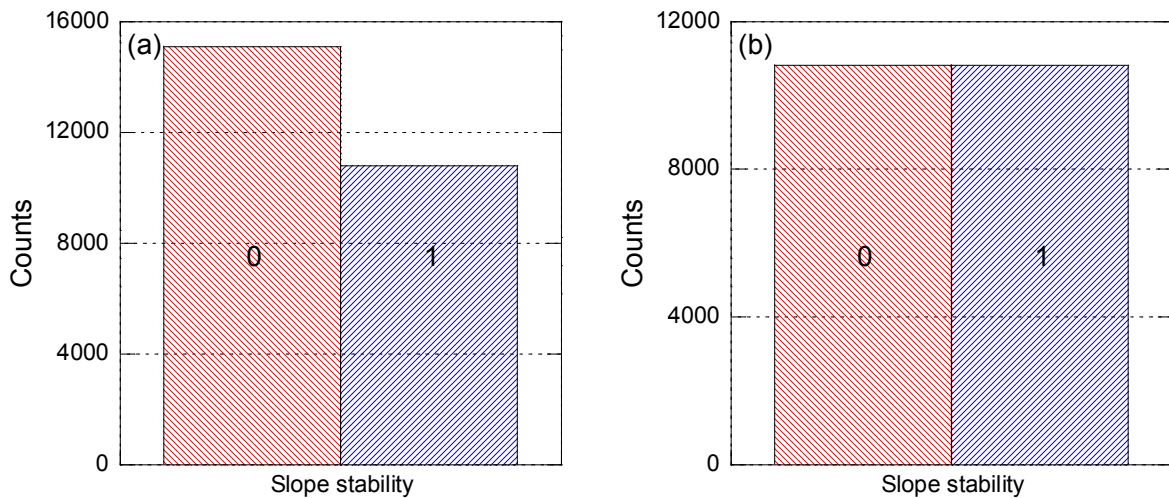


Figure 3-9. Adjusting the imbalanced target value (a) before adjusting about data imbalanced (b) after adjusting about data imbalanced.

Table 3-7. Variable names and types according to each analysis step

	No	Feature name	Data type (The number of index)	Analysis Case	
Input data	1	Slope length	numerical	Case-1, Case-3	
	2	Slope height	numerical	Case-1, Case-3	
	3	Slope angle	numerical	Case-1, Case-3	
	4	Slope gradient	numerical	Case-1, Case-3	
	5	Angle of upper slope	numerical	Case-1, Case-3	
	6	Distance from road	numerical	Case-1, Case-3	
	7	The number of berm	numerical	Case-1, Case-3	
	8	The number of valley	numerical	Case-1, Case-3	
	9	Soil depth	numerical	Case-1, Case-3	
	10	Slope type	categorical (3)	Case-2, Case-3	
	11	Topography	categorical (4)	Case-2, Case-3	
	12	Groundwater	categorical (5)	Case-2, Case-3	
	13	Leakage in vertical direction	categorical (4)	Case-2, Case-3	
	14	Leakage in horizontal direction	categorical (4)	Case-2, Case-3	
	15	Weathering	categorical (6)	Case-2, Case-3	
	16	Direction of discontinuities	categorical (4)	Case-2, Case-3	
	17	Slope shape	categorical (4)	Case-2, Case-3	
	18	Side shape of slope	categorical (4)	Case-2, Case-3	
	19	Lithology	categorical (4)	Case-2, Case-3	
	20	Bedrock shape	categorical (6)	Case-2, Case-3	
	21	Type of discontinuities	categorical (5)	Case-2, Case-3	
	22	Stability method	Type	categorical (4)	Case-2, Case-3
	23		State	categorical (5)	Case-2, Case-3
	24		Type	categorical (4)	Case-2, Case-3
	25		State	categorical (5)	Case-2, Case-3
	26		Type	categorical (4)	Case-2, Case-3
	27		State	categorical (5)	Case-2, Case-3
	28		Type	categorical (4)	Case-2, Case-3
	29		State	categorical (5)	Case-2, Case-3
	30		Type	categorical (4)	Case-2, Case-3
	31		State	categorical (5)	Case-2, Case-3
Output data (Target value)	Slope stability		binary (2)	Case-1, Case-2, Case-3	

4. 연구 결과

4.1 XGBoost 예측 모델

4.1.1 XGBoost 비탈면 안정성 예측 모델 초매개변수

XGBoost를 활용한 비탈면 안정성 예측 모델을 평가하기 위하여 훈련데이터와 시험데이터 비율 7:3으로 설정하였다. 초매개변수 최적화는 2.1.3절에서 설명한 베이지안 최적화 방법을 사용하였다. 훈련데이터에 대한 과적합을 방지하기 위하여 계층별(Stratified) k-fold 교차 검증은 5개로 분할하여 실시하였다. 그 결과를 정리하면 Table 4-1과 같다.

Table 4-1. Results of hyperparameter tuning for XGBoost according to each analysis step

	Case-1	Case-2	Case-3
booster	gbtree	gbtree	gbtree
colsample_bylevel	0.36351	0.41347	0.11039
colsample_bytree	0.60658	0.84282	0.87846
eval_metric	logloss	logloss	logloss
gamma	1.80801	1.52439	4.97449
learning_rate	0.44531	0.09860	0.13989
max_delta_step	0.03875	3.32170	7.87089
max_depth	4.29314	14.71294	4.36912
min_child_weight	83.20334	51.28230	30.73817
n_estimators	2252.502226	7966.31419	7645.546
objective	binary:logistic	binary:logistic	binary:logistic
reg_alpha	9.88183	3.20082	9.83848
reg_lambda	2.65249	9.62206	4.12595
scale_pos_weight	2.14001	1.37892	1.81335
subsample	0.31884	0.459430	0.34066

4.1.2 XGBoost 비탈면 안정성 예측 모델 : Case-1

수치형 데이터 9개로 구성된 Case-1과 XGBoost를 이용한 비탈면 안정성 예측 모델에 대해 학습 성능, AUC, 혼동행렬을 사용해 평가하였다. 모델 학습에 대한 학습 성능 평가는 Logloss를 이용하였다. Logloss는 분류에서 평가지표로 사용하는 지표 중 하나이며 식 (4.1)을 이용하여 구하며, 그 값이 0에 가까울수록 모델의 예측 성능이 좋음을 의미한다.

$$\log \frac{L(A)}{L(B)} = \log L(A) - \log L(B) = l(A) - l(B) \quad (4.1)$$

학습 성능 평가 결과, 훈련데이터와 시험데이터가 유사한 성능을 나타냈으며 과적합 현상은 발생하지 않았다(Figure 4-1). 분류 성능 지표로 많이 사용되는 AUC의 경우는 0.668로 산출되었다(Figure 4-2). 훈련 및 시험데이터에 대한 출력데이터의 실제값과 예측값에 대한 빈도는 Figure 4-3과 같이 나타났으며, 이를 통해 산출한 정밀도, 재현율, F1-Score, 정확도의 결과는 다음과 같다(Table 4-2).

훈련데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 정밀도는 70.0%, 안정으로 예측한 경우의 정밀도는 64.0%, 평균 정밀도는 68.0%로 나타났다. 시험데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 정밀도는 69.0%, 안정으로 예측한 경우의 정밀도는 65.0%, 평균 정밀도는 67.0%로 나타났다.

훈련데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 재현율은 61.0%, 안정으로 예측한 경우의 재현율은 74.0%, 평균 재현율은 67.5%로 나타났다. 시험데이터를 사용한 예측 모델에서 비탈면의 상태를 불안정으로 예측한 경우의 재현율은 61.0%, 안정으로 예측한 경우의 재현율은 73.0%, 평균 재현율은 67.0%로 나타났다.

훈련데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 F1-Score는 66.0%, 안정으로 예측한 경우의 F1-Score는 70.0%, 평균 F1-Score는 68.0%로 나타났다. 시험데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 F1-Score는 65.0%, 안정으로 예측한 경우의 F1-Score는 67.0%, 평균 F1-Score는 66.0%로 나타났다.

트리를 분할할 때 사용되는 변수 중요도는 ‘사면길이’, ‘사면각도’, ‘토층심도’ 순으로 나타났으며, ‘소단 개소’, ‘계곡부 개소’를 제외하고 대부분 큰 비중을 차지하였다(Figure 4-4).

Table 4-2. Performance of Case-1 as a result of classification using XGBoost

		Precision(%)	Recall(%)	F1-Score(%)	Accuracy(%)
Prediction performance for train data	Unstable (Label : 0)	70.0	61.0	66.0	68.0
	Stable (Label : 1)	66.0	74.0	70.0	
	Average	68.0	67.5	68.0	
Prediction performance for test data	Unstable (Label : 0)	69.0	61.0	65.0	67.0
	Stable (Label : 1)	65.0	73.0	67.0	
	Average	67.0	67.0	66.0	

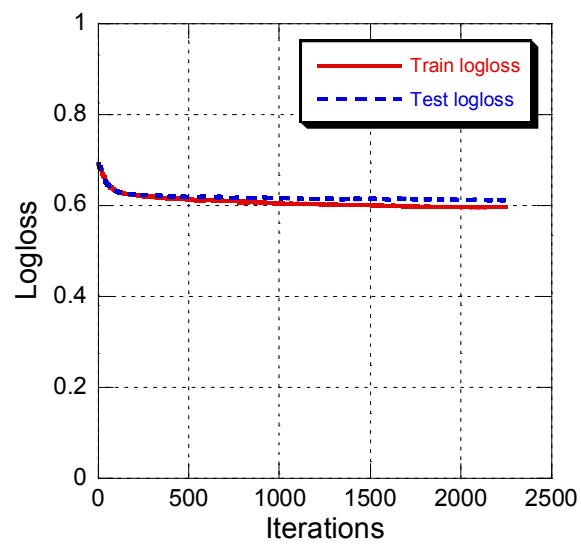


Figure 4-1. Result for prediction performance of Case-1 using XGBoost.

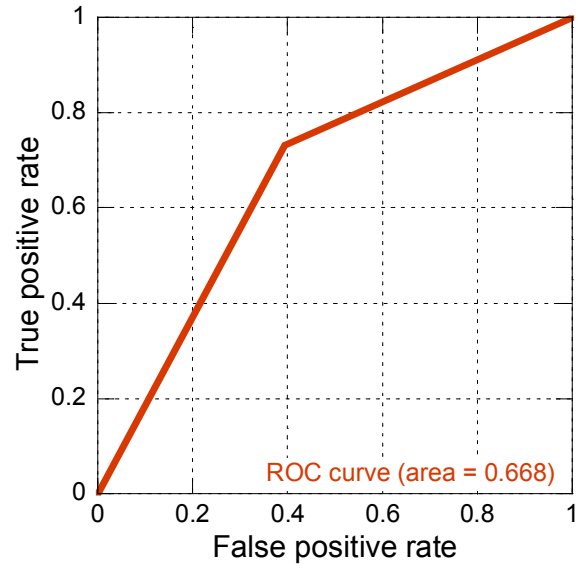


Figure 4-2. Result for AUC of Case-1 using XGBoost.

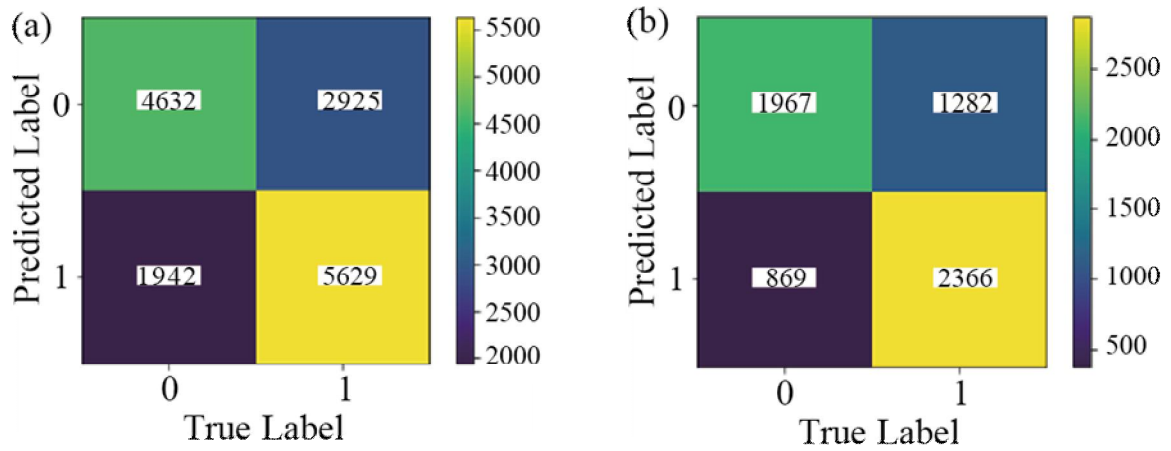


Figure 4-3. Results for confusion matrix of Case-1 on (a) train data and (b) test data using XGBoost.

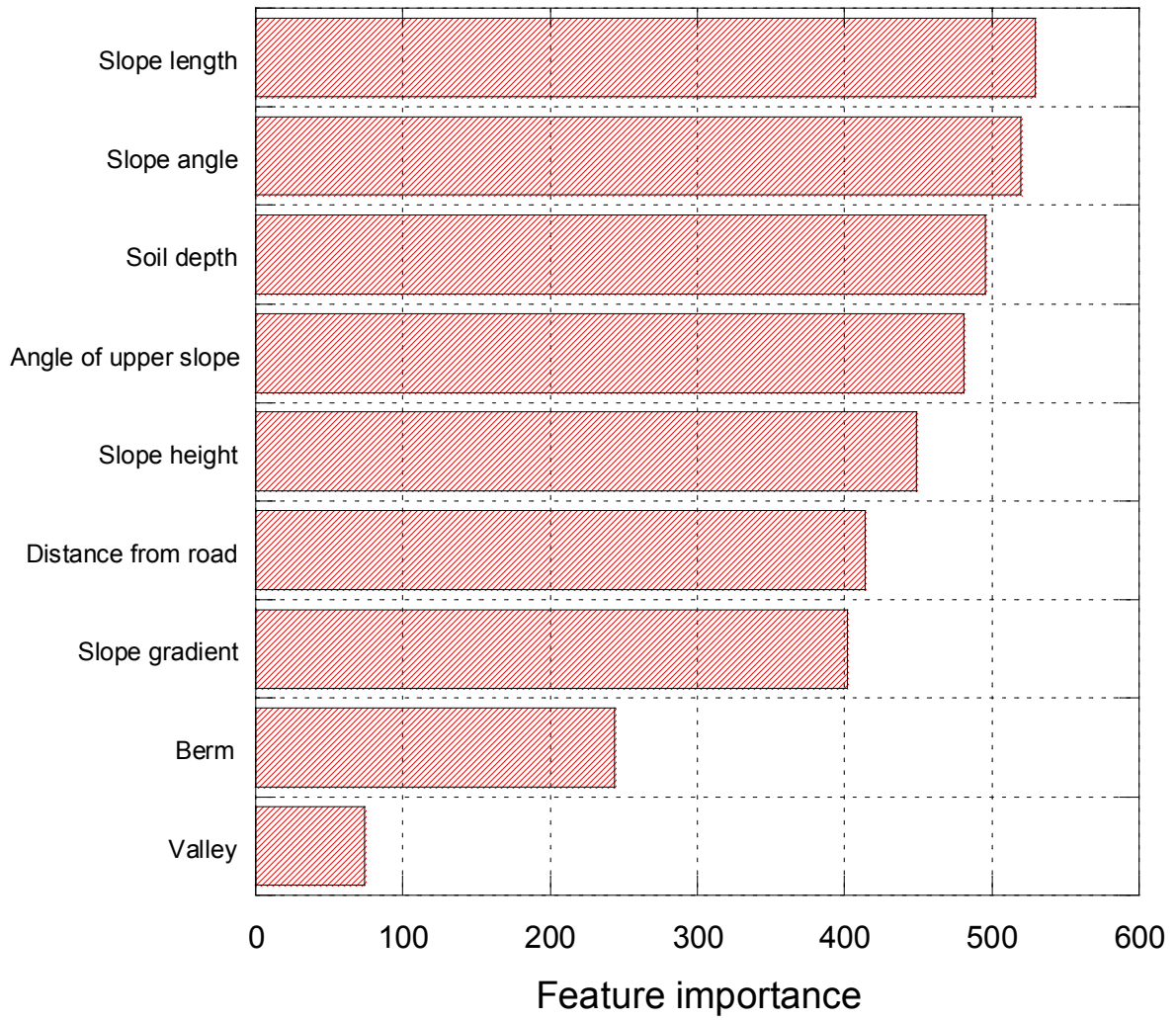


Figure 4-4. Result for feature importance of Case-1 using XGBoost.

4.1.3 XGBoost 비탈면 안정성 예측 모델 : Case-2

범주형 데이터 22개와 XGBoost를 이용한 비탈면 안정성 예측 모델에 대해 학습 성능, AUC, 혼동행렬을 사용해 평가하였다. 모델 학습에 대한 학습 성능 평가는 Logloss를 이용하였다. 그 결과에 의하면, 모델 성능은 학습데이터에 약한 과적합을 보이는 것으로 나타났다(Figure 4-5). 분류 성능 지표로 많이 사용되는 AUC의 경우는 0.771로 산출되었다(Figure 4-6). 훈련 및 시험데이터에 대한 출력데이터의 실제값과 예측값에 대한 빈도는 Figure 4-7과 같이 나타났으며, 이를 통해 산출한 정밀도, 재현율, F1-Score, 정확도의 결과는 다음과 같다(Table 4-3).

훈련데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 정밀도는 87.0%, 안정으로 예측한 경우의 정밀도는 79.0%, 평균 정밀도는 83.0%로 나타났다. 시험데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 정밀도는 81.0%, 안정으로 예측한 경우의 정밀도는 74.0%, 평균 정밀도는 77.5%로 나타났다.

훈련데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 재현율은 76.0%, 안정으로 예측한 경우의 재현율은 89.0%, 평균 재현율은 82.5%로 나타났다. 시험데이터를 사용한 예측 모델에서 비탈면의 상태를 불안정으로 예측한 경우의 재현율은 70.0%, 안정으로 예측한 경우의 재현율은 84.0%, 평균 재현율은 77.0%로 나타났다.

훈련데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 F1-Score는 82.0%, 안정으로 예측한 경우의 F1-Score는 84.0%, 평균 F1-Score는 83.0%로 나타났다. 시험데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 F1-Score는 75.0%, 안정으로 예측한 경우의 F1-Score는 79.0%, 평균 F1-Score는 77.0%로 나타났다.

트리를 분할할 때 사용되는 변수 중요도는 ‘주변지형’, ‘암종’, ‘암반형태’, ‘비탈면 보강공법 2번 공법 종류’, ‘풍화도’가 다른 변수들보다 상대적으로 높게 나타났다(Figure 4-8).

Table 4-3. Performance of Case-2 as a result of classification using XGBoost

		Precision(%)	Recall(%)	F1-Score(%)	Accuracy(%)
Prediction performance for train data	Unstable (Label : 0)	87.0	76.0	82.0	83.0
	Stable (Label : 1)	79.0	89.0	84.0	
	Average	83.0	82.5	83.0	
Prediction performance for test data	Unstable (Label : 0)	81.0	70.0	75.0	77.0
	Stable (Label : 1)	74.0	84.0	79.0	
	Average	77.5	77.0	77.0	

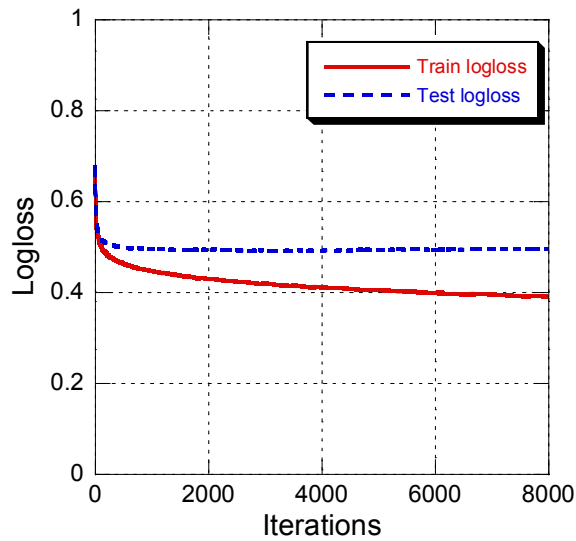


Figure 4-5. Result for prediction performance of Case-2 using XGBoost.

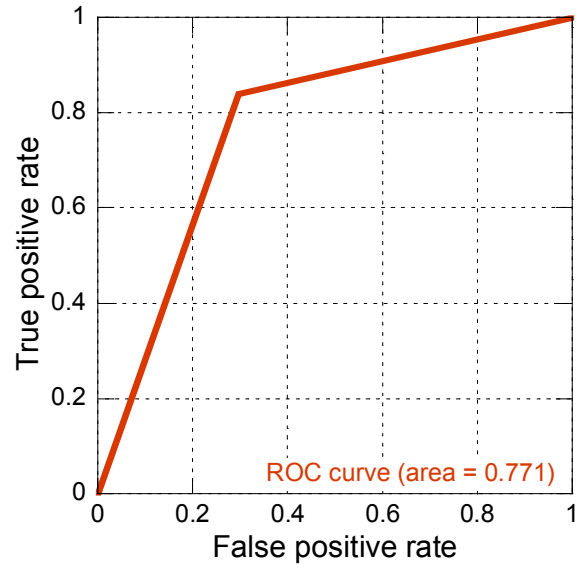


Figure 4-6. Result for AUC of Case-2 using XGBoost.

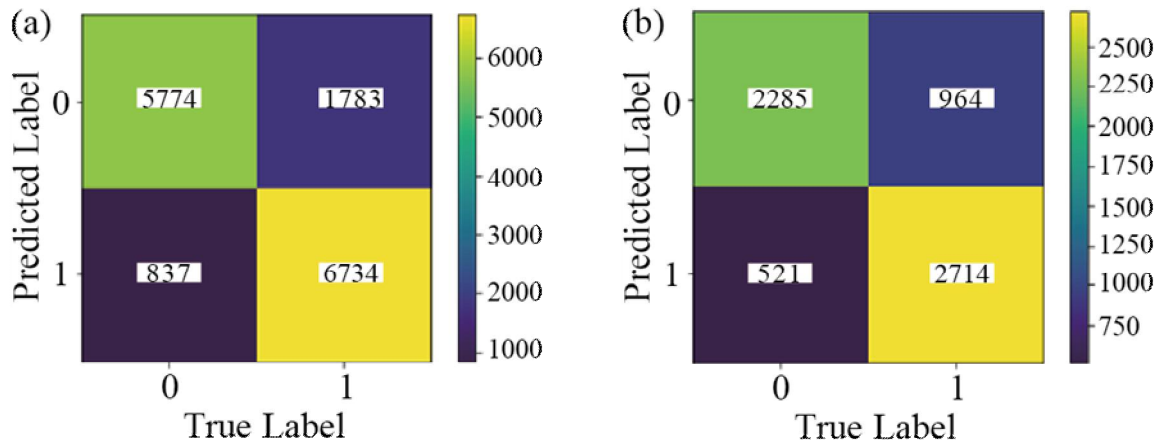


Figure 4-7. Results for confusion matrix of Case-2 on (a) train data and (b) test data using XGBoost

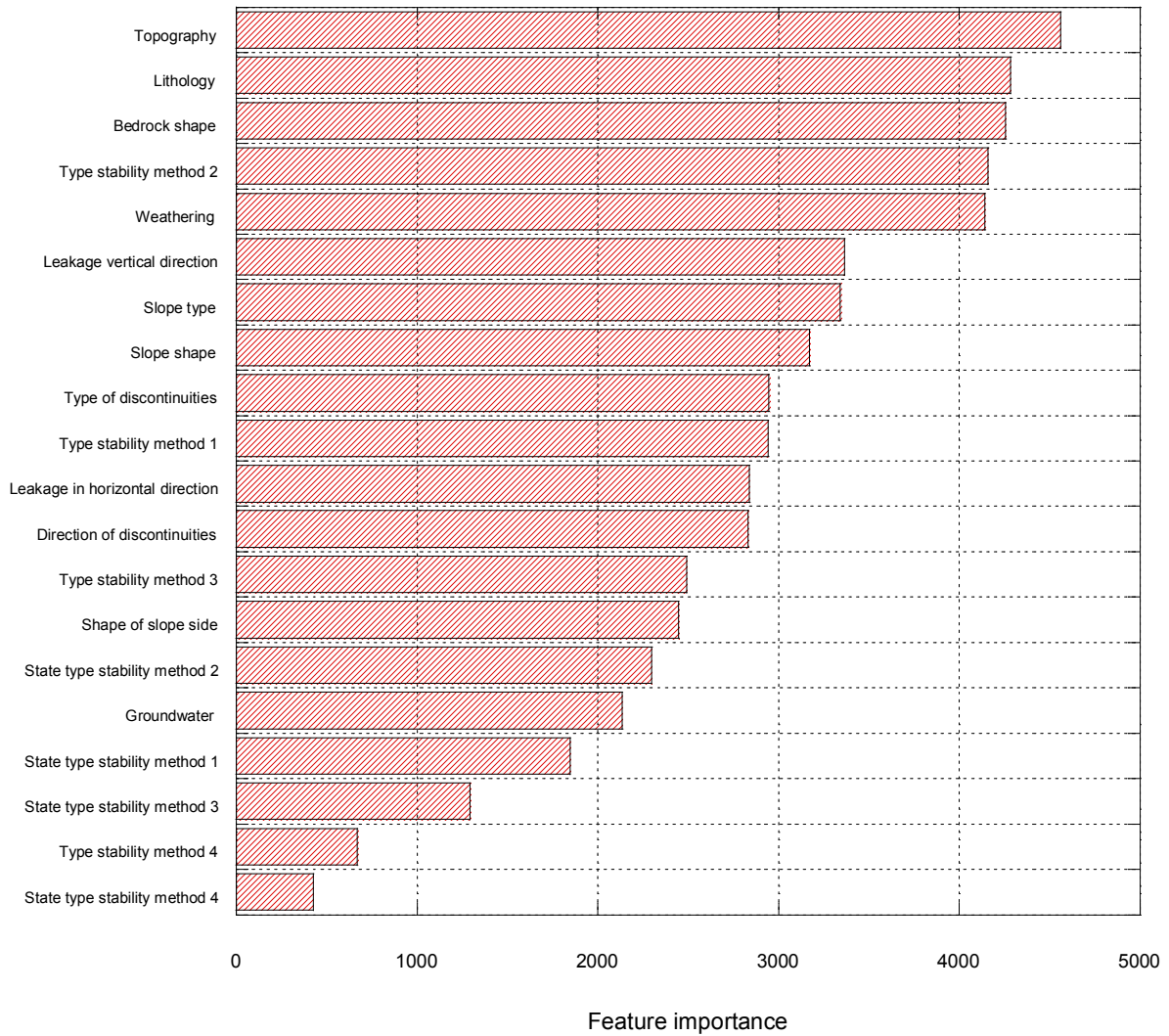


Figure 4-8. Result for feature importance of Case-2 using XGBoost.

4.1.4 XGBoost 비탈면 안정성 예측 모델 : Case-3

수치형과 범주형 데이터를 포함한 31개의 입력데이터와 XGBoost을 이용한 비탈면 안정성 예측 모델에 대해 학습 성능, AUC, 혼동행렬을 사용해 평가하였다. 모델 학습에 대한 학습 성능 평가는 Logloss를 이용하였다. 그 결과에 의하면, 모델 성능은 Case-2와 비슷하게 학습데이터에 약한 과적합을 보였다(Figure 4-9). 분류 성능 지표로 많이 사용되는 AUC의 경우는 0.770로 산출되었다(Figure 4-10). 훈련 및 시험데이터에 대한 출력데이터의 실제값과 예측값에 대한 빈도는 Figure 4-11과 같이 나타났으며, 이를 통해 산출한 정밀도, 재현율, F1-Score, 정확도의 결과는 다음과 같다(Table 4-4).

훈련데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 정밀도는 89.0%, 안정으로 예측한 경우의 정밀도는 76.0%, 평균 정밀도는 82.5.0%로 나타났다. 시험데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 정밀도는 85.0%, 안정으로 예측한 경우의 정밀도는 72.0%, 평균 정밀도는 78.5%로 나타났다.

훈련데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 재현율은 71.0%, 안정으로 예측한 경우의 재현율은 92.0%, 평균 재현율은 81.5%로 나타났다. 시험데이터를 사용한 예측 모델에서 비탈면의 상태를 불안정으로 예측한 경우의 재현율은 66.0%, 안정으로 예측한 경우의 재현율은 88.0%, 평균 재현율은 77.0%로 나타났다.

훈련데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 F1-Score는 79.0%, 안정으로 예측한 경우의 F1-Score는 83.0%, 평균 F1-Score는 81.0%로 나타났다. 시험데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 F1-Score는 74.0%, 안정으로 예측한 경우의 F1-Score는 79.0%, 평균 F1-Score는 76.5%로 나타났다.

트리를 분할할 때 사용되는 변수의 중요도는 ‘사면길이’, ‘사면상부경사’, ‘토층심도’ 등 ‘소단 개소’, ‘계곡부 개소’를 제외한 수치형 데이터 7개가 다른 변수들에 비해 상대적으로 높게 나타났다(Figure 4-12).

Table 4-4. Performance of Case-3 as a result of classification using XGBoost

		Precision(%)	Recall(%)	F1-Score(%)	Accuracy(%)
Prediction performance for train data	Unstable (Label : 0)	89.0	71.0	79.0	81.0
	Stable (Label : 1)	76.0	92.0	83.0	
	Average	82.5	81.5	81.0	
Prediction performance for test data	Unstable (Label : 0)	85.0	66.0	74.0	77.0
	Stable (Label : 1)	72.0	88.0	79.0	
	Average	78.5	77.0	76.5	

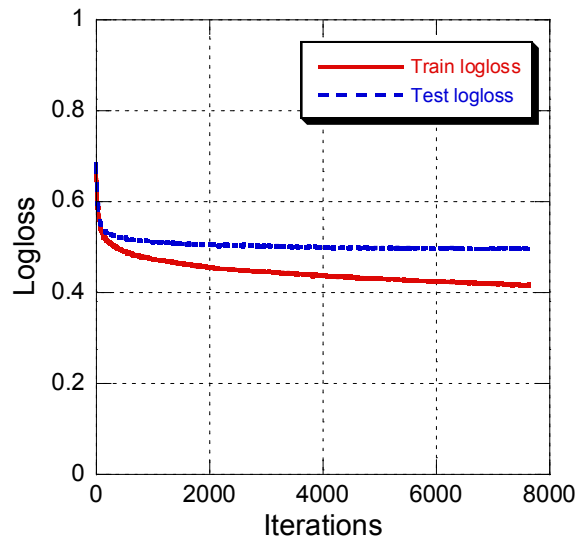


Figure 4-9. Result for prediction performance of Case-3 using XGBoost.

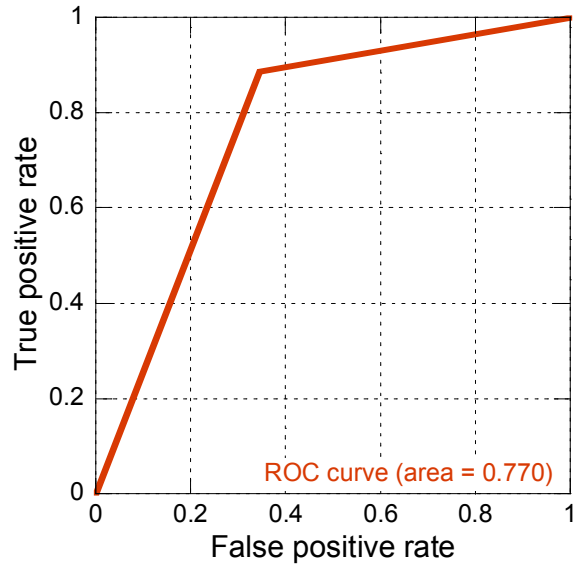


Figure 4-10. Result for AUC of Case-3 using XGBoost.

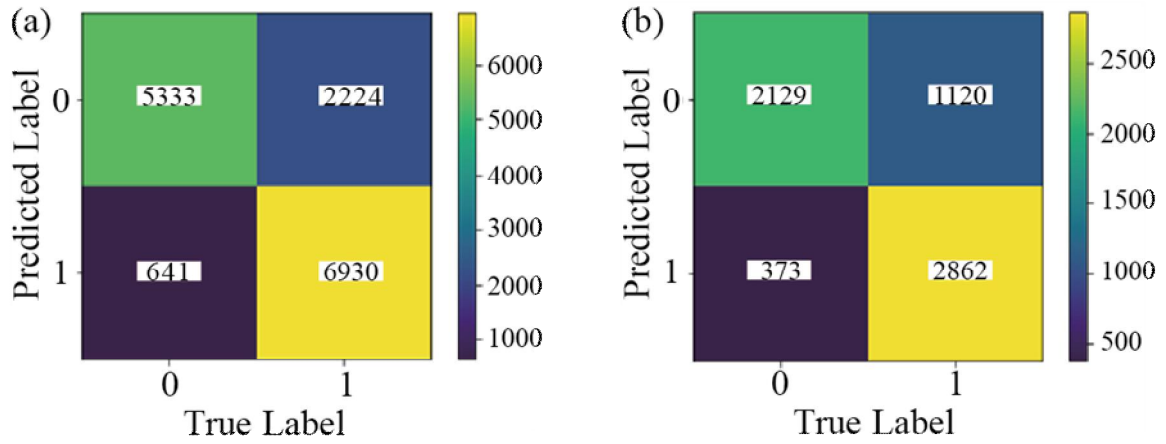


Figure 4-11. Results for confusion matrix of Case-3 on (a) train data and (b) test data using XGBoost.

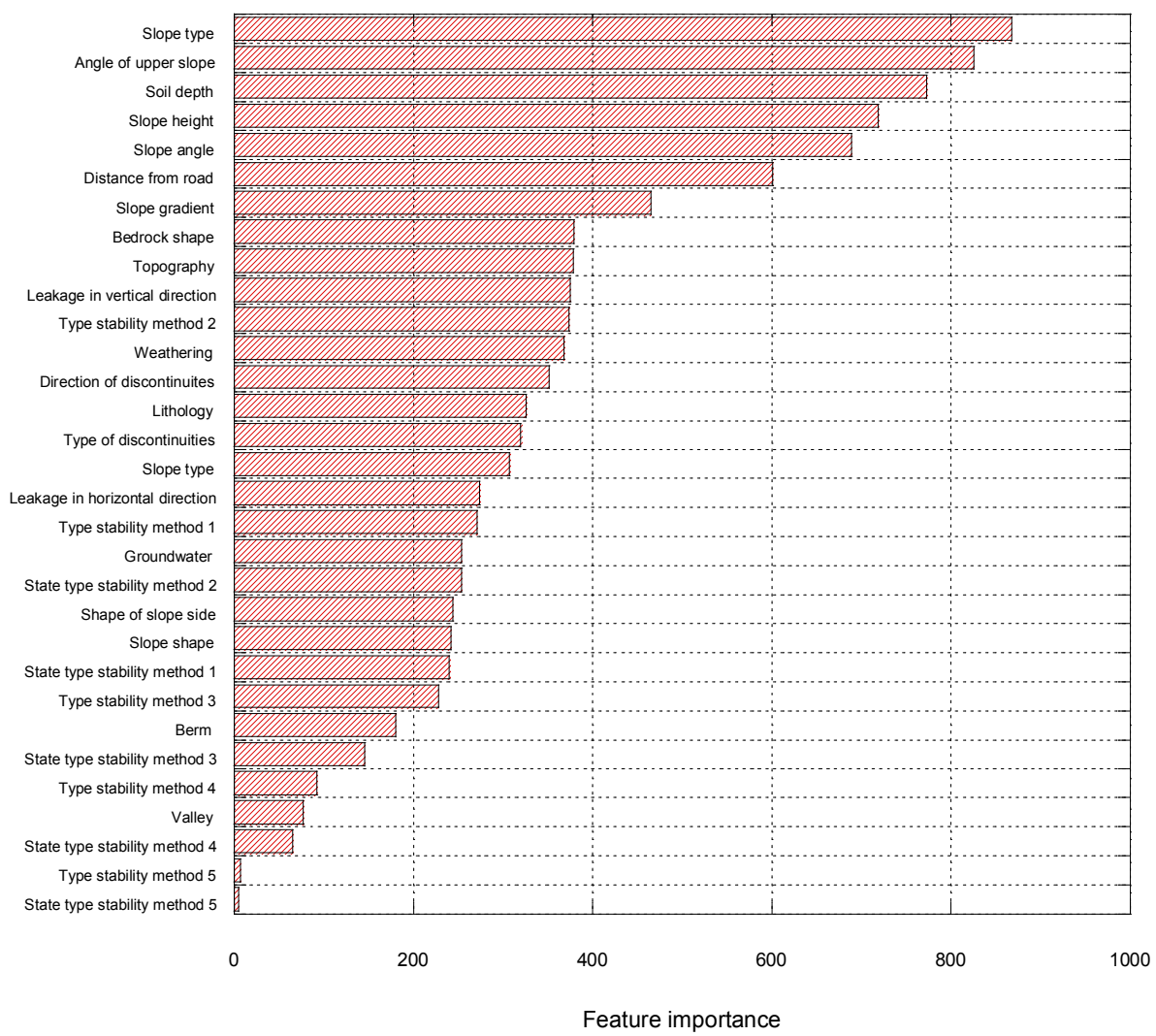


Figure 4-12. Result for feature importance of Case-3 using XGBoost.

4.1.5 XGBoost 비탈면 안정성 예측 모델 결과

비탈면 안정성 예측 모델을 제안하기 위해 XGBoost와 입력데이터에 따른 3가지의 해석 단계를 이용하여 각 모델의 결과를 산출하였다. 비탈면 안정성에 대한 각 변수의 영향을 비교 분석하기 위해 해석 단계를 수치형 데이터 9개로 구성된 Case-1, 범주형 데이터 22개로 구성된 Case-2, 수치형 데이터와 범주형 데이터 31개로 구성된 Case-3으로 구분하였다(Table 3-6). 훈련데이터와 시험데이터 비율은 7:3으로 설정하였고, 초매개변수 최적화는 베이지안 최적화 방법을 사용하였다. 훈련데이터에 대한 과적합을 방지하기 위하여 계층별 k-fold 교차 검증을 5분할 하여 실시하였다. 각 해석 단계에 따른 초매개변수 산정 결과는 Table 4-1과 같다. 모델 학습에 대한 평가는 Logloss를 이용하였고, 모델의 예측 성능 평가는 AUC, 정밀도, 재현율, F1-Score, 정확도를 사용하였다.

3가지 해석 단계에 대한 학습 성능 분석 결과, Case-2가 가장 좋은 성능을 보였고, 다음으로 Case-3, Case-1 순으로 나타났다. Case-1은 훈련데이터와 시험데이터가 유사한 학습 곡선을 나타냈지만, Case-2와 Case-3은 훈련데이터에 약한 과적합 현상이 나타났다. AUC는 Case-1의 경우 0.668, Case-2의 경우 0.771, Case-3의 경우 0.770으로 나타났다. AUC는 3가지의 해석 단계 중 Case-2가 성능이 가장 좋게 나타났고, 그다음 Case-3, Case-1 순으로 나타났다.

Case-1의 예측 성능 결과, 훈련데이터와 시험데이터 사이의 정밀도, 재현율, F1-Score, 그리고 정확도는 유사하게 나타났고, 출력데이터의 값에 따른 정밀도, 재현율, F1-Score는 훈련데이터와 시험데이터에서 4~12% 차이를 나타냈다. 또한 정밀도와 재현율은 훈련데이터와 시험데이터에서 8~9%의 예측 성능 불균형이 나타났다(Figure 4-13). Case-2의 예측 성능 결과, 훈련데이터와 시험데이터 사이의 정밀도, 재현율, F1-Score, 그리고 정확도는 약 5%의 차이를 보였고, 출력데이터의 값에 따른 정밀도, 재현율, F1-Score는 훈련데이터와 시험데이터에서 4~14% 차이를 나타냈다. 또한 정밀도와 재현율은 훈련데이터와 시험데이터에서 약 10~11%의 예측 성능 불균형이 나타났다(Figure 4-13). Case-2의 경우 Case-1 보다 예측 성능 불균형이 2% 더 크게 나타났다. Case-3의 예측 성능 결과, 훈련데이터와 시험데이터 사이의 정밀도, 재현율, F1-Score, 그리고 정확도는 약 5%의 차이를 보였고, 출력데이터의 값에 따른 정밀도, 재현율, F1-Score는 훈련데이터와 시험데이터

에서 4~22% 차이를 보였다. 또한 정밀도와 재현율은 훈련데이터와 시험데이터에서 약 16~19%의 차이를 보이는 예측 성능 불균형이 나타났다(Figure 4-13). 예측 성능의 결과를 종합하면 해석 단계에 따른 예측 성능은 Case-2가 가장 좋게 나타났고, 그다음 Case-3, Case-1 순으로 높게 나타났다. 각 해석 단계에 따른 예측 결과 성능 불균형은 범주형 데이터가 포함된 Case-2와 Case-3에서 크게 발생하였다. 이는 XGBoost가 사용하는 트리 분할 방식과 데이터 형태에 따른 데이터 전처리의 차이로 인한 것으로 사료 된다. 수치형 데이터는 이상치를 제거해 정규분포형태를 띄고 있는 반면에, 범주형 데이터는 각각의 변수마다 편향되어 있기 때문이다. 또한 예측 성능이 높은 만큼 훈련데이터에 과적합 하려는 현상과 예측 불균형의 증가하는 추세가 함께 나타났으며 이를 통해 전문가에 의해 조사된 주관적 자료인 범주형 데이터가 모델 예측 성능에 있어 중요한 자료인 동시에 훈련데이터에 과적합을 발생시키는 자료이며, 예측 오류를 증가시키는 요인으로 판단된다.

Case-1의 변수 중요도는 ‘소단 개소’와 ‘계곡부 개소’를 제외한 ‘사면길이’, ‘사면경사’ 등의 변수들은 비슷한 중요도를 나타냈다(Figure 4-4). Case-2의 변수 중요도는 ‘주변지형’, ‘암종’, ‘암반형태’, ‘비탈면 보강공법 2의 공법 종류’, ‘풍화도’가 다른 변수들보다 상대적으로 높게 나타났다(Figure 4-8). Case-3의 변수 중요도는 ‘사면종류’가 가장 높게 나타났고, 그다음 ‘사면길이’, ‘사면상부경사’, ‘토층심도’ 등 ‘소단 개소’와 ‘계곡부 개소’를 제외한 수치형 데이터가 높게 나타났다(Figure 4-12). XGBoost를 활용한 비탈면 안정성 예측 모델의 변수 중요도는 전반적으로 범주형 데이터보다 수치형 데이터가 상대적으로 높게 나타났다. 수치형 데이터 중에서 ‘사면상부경사’, ‘토층심도’, ‘사면각도’의 변수 중요도가 높게 나타났으며, 범주형 데이터 중에서는 ‘암반형태’, ‘주변지형’, ‘지하수’의 변수 중요도가 높게 나타났다.

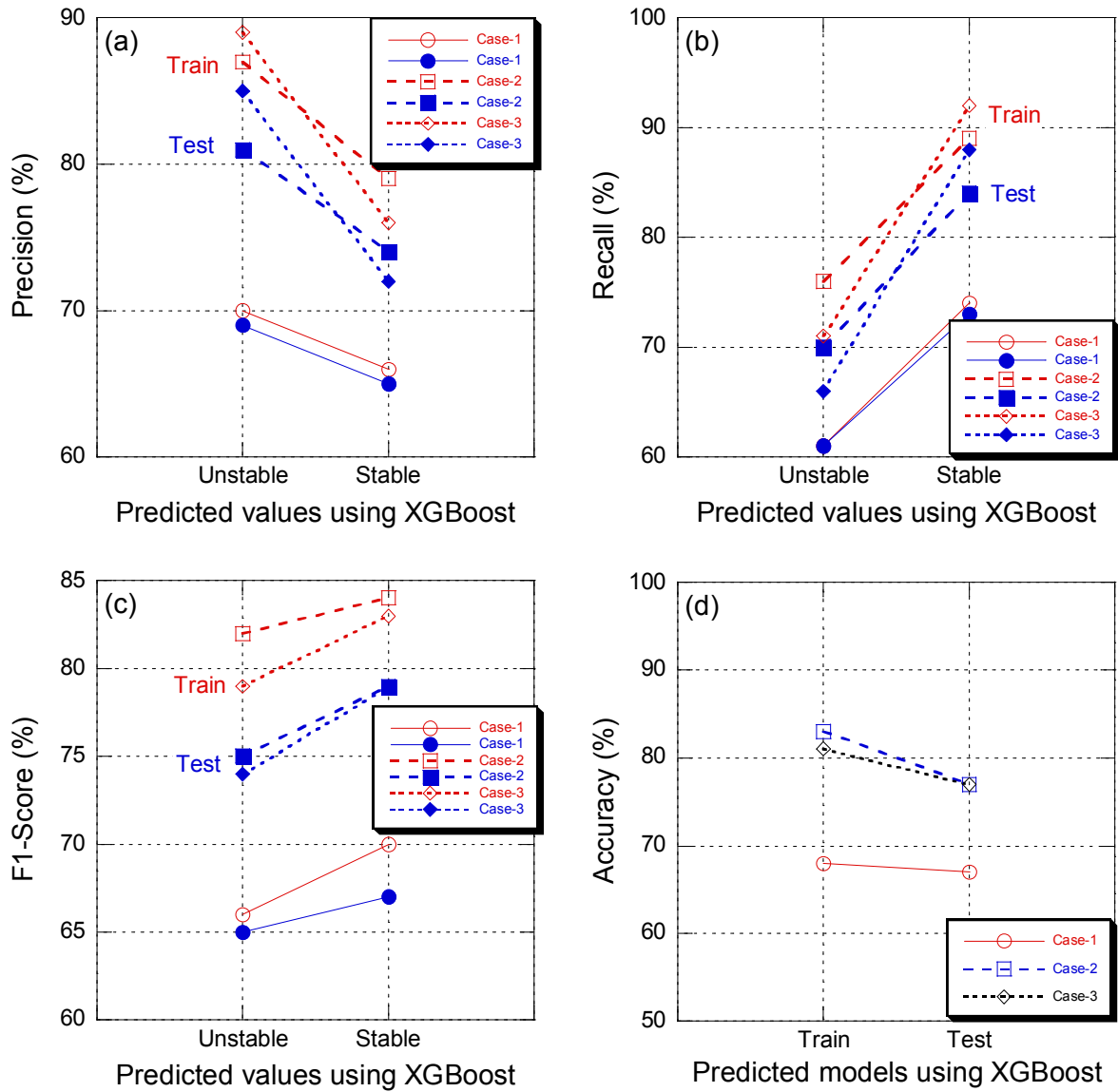


Figure 4-13. Result for slope stability prediction model using XGBoost (a) precision, (b) recall (c) F1-Score, (d) accuracy.

4.2 LightGBM 예측 모델

4.2.1 LightGBM 비탈면 안정성 예측 모델 초매개변수

LightGBM을 활용한 비탈면 안정성 예측 모델을 평가하기 위하여 훈련데이터와 시험데이터 비율 7:3으로 설정하였다. 초매개변수 최적화는 2.1.3절에서 설명한 베이지안 최적화 방법을 사용하였다. 훈련데이터에 대한 과적합을 방지하기 위하여 계층별 k-fold 교차검증은 5개 분할하여 실시하였다. 그 결과를 정리하면, Table 4-5와 같다. 다른 모델들과 다르게 LightGBM의 초매개변수는 Case-1, Case-2, 그리고 Case-3 모두에서 동일한 초매개변수 값이 산출되었다.

Table 4-5. Results of hyperparameter tuning for LightGBM according to each analysis step

	Case-1	Case-2	Case-3
boosting_type	gbdt	gbdt	gbdt
colsample_bytree	0.781015	0.781015	0.781015
eval_metric	neg_log_loss	neg_log_loss	neg_log_loss
learning_rate	0.015960	0.015960	0.015960
max_depth	14.362484	14.362484	14.362484
min_child_samples	18.1188758	18.1188758	18.1188758
min_child_weight	0.5185915	0.5185915	0.5185915
min_split_gain	0.66310916	0.66310916	0.66310916
n_estimators	1612.7317	1612.7317	1612.7317
num_leaves	181.265637	181.265637	181.265637
objective	binary	binary	binary
reg_alpha	2.942950	2.942950	2.942950
reg_lambda	3.26875	3.26875	3.26875
subsample	0.15830287	0.15830287	0.15830287
subsample_for_bin	370736	370736	370736

4.2.2 LightGBM 비탈면 안정성 예측 모델 : Case-1

수치형 데이터 9개와 LightGBM을 이용한 비탈면 안정성 예측 모델에 대해 학습 성능, AUC, 혼동행렬을 사용해 평가하였다. 모델 학습에 대한 학습 성능 평가는 Logloss를 이용하였다. 그 결과에 의하면, 훈련데이터와 시험데이터가 유사한 성능을 나타냈으며 과적합 현상은 발생하지 않았다(Figure 4-14). 분류 성능 지표로 많이 사용되는 AUC의 경우는 0.661로 산출되었다(Figure 4-15). 훈련 및 시험데이터에 대한 출력데이터의 실제값과 예측값에 대한 빈도는 Figure 4-16과 같이 나타났으며, 이를 통해 산출한 정밀도, 재현율, F1-Score, 정확도의 결과는 다음과 같다(Table 4-6).

훈련데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 정밀도는 74.0%, 안정으로 예측한 경우의 정밀도는 72.0%, 평균 정밀도는 73.0%로 나타났다. 시험데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 정밀도는 69.0%, 안정으로 예측한 경우의 정밀도는 68.0%, 평균 정밀도는 68.5%로 나타났다.

훈련데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 재현율은 72.0%, 안정으로 예측한 경우의 재현율은 74.0%, 평균 재현율은 73.0%로 나타났다. 시험데이터를 사용한 예측 모델에서 비탈면의 상태를 불안정으로 예측한 경우의 재현율은 67.0%, 안정으로 예측한 경우의 재현율은 69.0%, 평균 재현율은 68.0%로 나타났다.

훈련데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 F1-Score는 73.0%, 안정으로 예측한 경우의 F1-Score는 73.0%, 평균 F1-Score는 73.0%로 나타났다. 시험데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 F1-Score는 68.0%, 안정으로 예측한 경우의 F1-Score는 68.0%, 평균 F1-Score는 68.0%로 나타났다.

트리를 분할할 때 사용되는 변수 중요도는 ‘사면길이’, ‘사면상부경사’, ‘사면경사’가 다른 변수들에 비해 상대적으로 큰 비중을 차지한 것으로 나타났다(Figure 4-17).

Table 4-6. Performance of Case-1 as a result of classification using LightGBM

		Precision(%)	Recall(%)	F1-Score(%)	Accuracy(%)
Prediction performance for train data	Unstable (Label : 0)	74.0	72.0	73.0	73.0
	Stable (Label : 1)	72.0	74.0	73.0	
	Average	73.0	73.0	73.0	
Prediction performance for test data	Unstable (Label : 0)	69.0	67.0	68.0	68.0
	Stable (Label : 1)	68.0	69.0	68.0	
	Average	68.5	68.0	68.0	

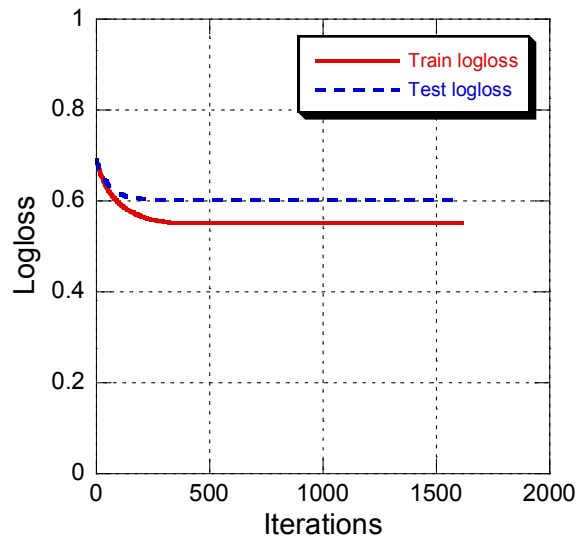


Figure 4-14. Result for prediction performance of Case-1 using LightGBM.

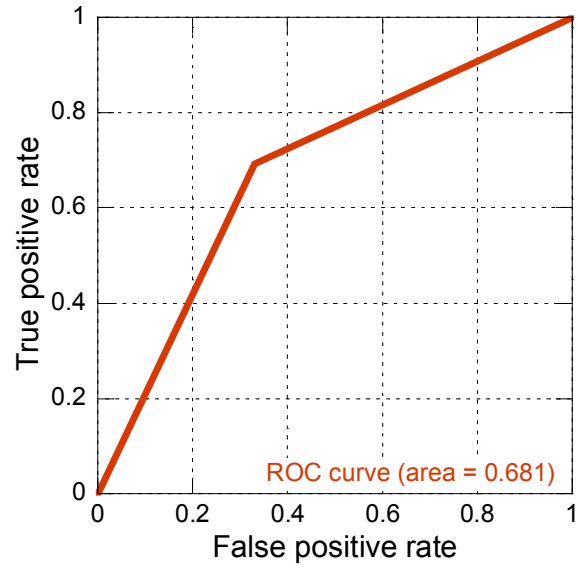


Figure 4-15. Result for AUC of Case-1 using LightGBM.

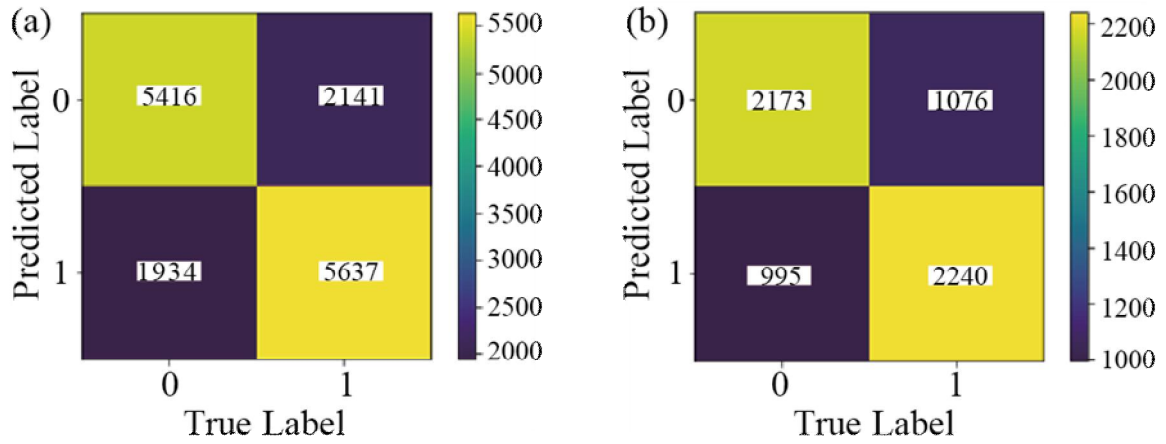


Figure 4-16. Results for confusion matrix of Case-1 on (a) train data and (b) test data using LightGBM.

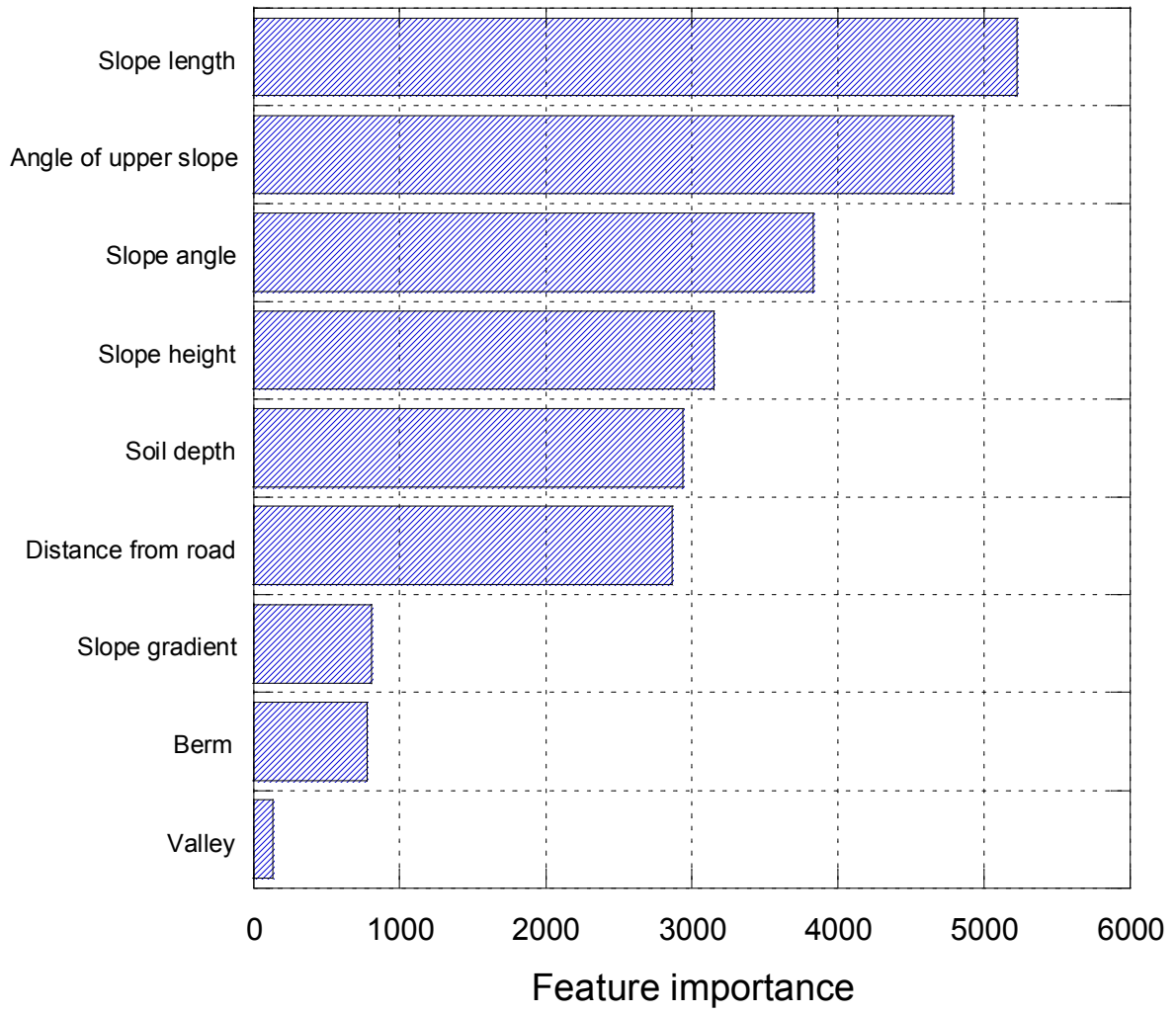


Figure 4-17. Result for feature importance of Case-1 using LightGBM.

4.2.3 LightGBM 비탈면 안정성 예측 모델 : Case-2

범주형 데이터 22개와 LightGBM을 이용한 비탈면 안정성 예측 모델에 대해 학습 성능, AUC, 혼동행렬을 사용해 평가하였다. 모델 학습에 대한 학습 성능 평가는 Logloss를 이용하였다. 그 결과에 의하면, 학습데이터와 시험데이터에서 유사한 성능을 보이는 것으로 나타났으며 과적합 현상은 발생하지 않았다(Figure 4-18). 분류 성능 지표로 많이 사용되는 AUC의 경우는 0.766로 산출되었다(Figure 4-19). 훈련 및 시험데이터에 대한 출력데이터의 실제값과 예측값에 대한 빈도는 Figure 4-20과 같이 나타났으며, 이를 통해 산출한 정밀도, 재현율, F1-Score, 정확도의 결과는 다음과 같다(Table 4-7).

훈련데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 정밀도는 80.0%, 안정으로 예측한 경우의 정밀도는 78.0%, 평균 정밀도는 79.0%로 나타났다. 시험데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 정밀도는 78.0%, 안정으로 예측한 경우의 정밀도는 75.0%, 평균 정밀도는 76.5%로 나타났다.

훈련데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 재현율은 78.0%, 안정으로 예측한 경우의 재현율은 81.5%, 평균은 79.5%로 나타났다. 시험데이터를 사용한 예측 모델에서 비탈면의 상태를 불안정으로 예측한 경우의 재현율은 75.0%, 안정으로 예측한 경우의 재현율은 79.0%, 평균 재현율은 77.0%로 나타났다.

훈련데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 F1-Score는 79.0%, 안정으로 예측한 경우의 F1-Score는 80.0%, 평균 F1-Score는 79.5%로 나타났다. 시험데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 F1-Score는 76.0%, 안정으로 예측한 경우의 F1-Score는 77.0%, 평균 F1-Score는 76.5%로 나타났다.

트리를 분할할 때 사용되는 변수 중요도는 ‘암반형태’, ‘주변지형’의 변수들이 다른 변수들보다 상대적으로 크게 나타났다(Figure 4-21).

Table 4-7. Performance of Case-2 as a result of classification using LightGBM

		Precision(%)	Recall(%)	F1-Score(%)	Accuracy(%)
Prediction performance for train data	Unstable (Label : 0)	80.0	78.0	79.0	79.0
	Stable (Label : 1)	78.0	81.0	80.0	
	Average	79.0	79.5	79.5	
Prediction performance for test data	Unstable (Label : 0)	78.0	75.0	76.0	77.0
	Stable (Label : 1)	75.0	79.0	77.0	
	Average	76.5	77.0	76.5	

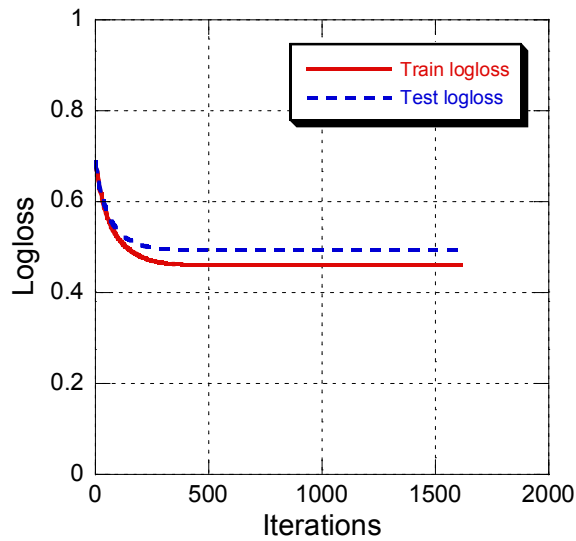


Figure 4-18. Result for prediction performance of Case-2 using LightGBM.

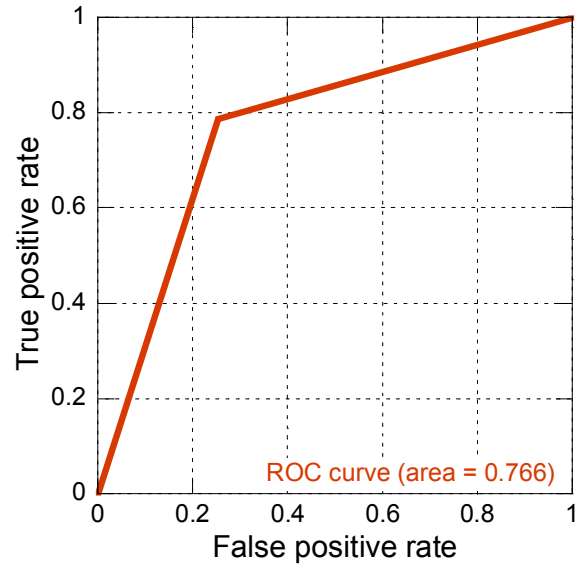


Figure 4-19. Result for AUC of Case-2 using LightGBM.

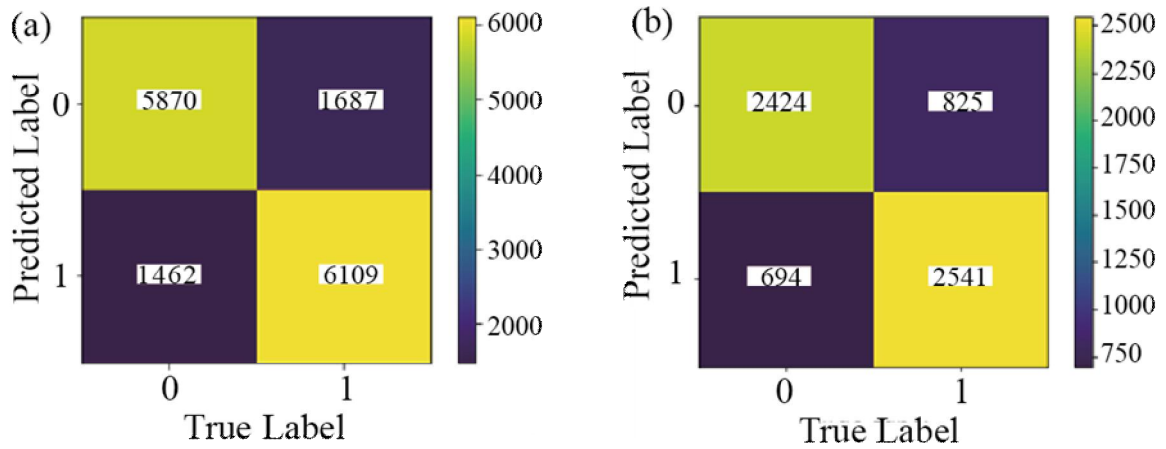


Figure 4-20. Results for confusion matrix of Case-2 on (a) train data and (b) test data using LightGBM.

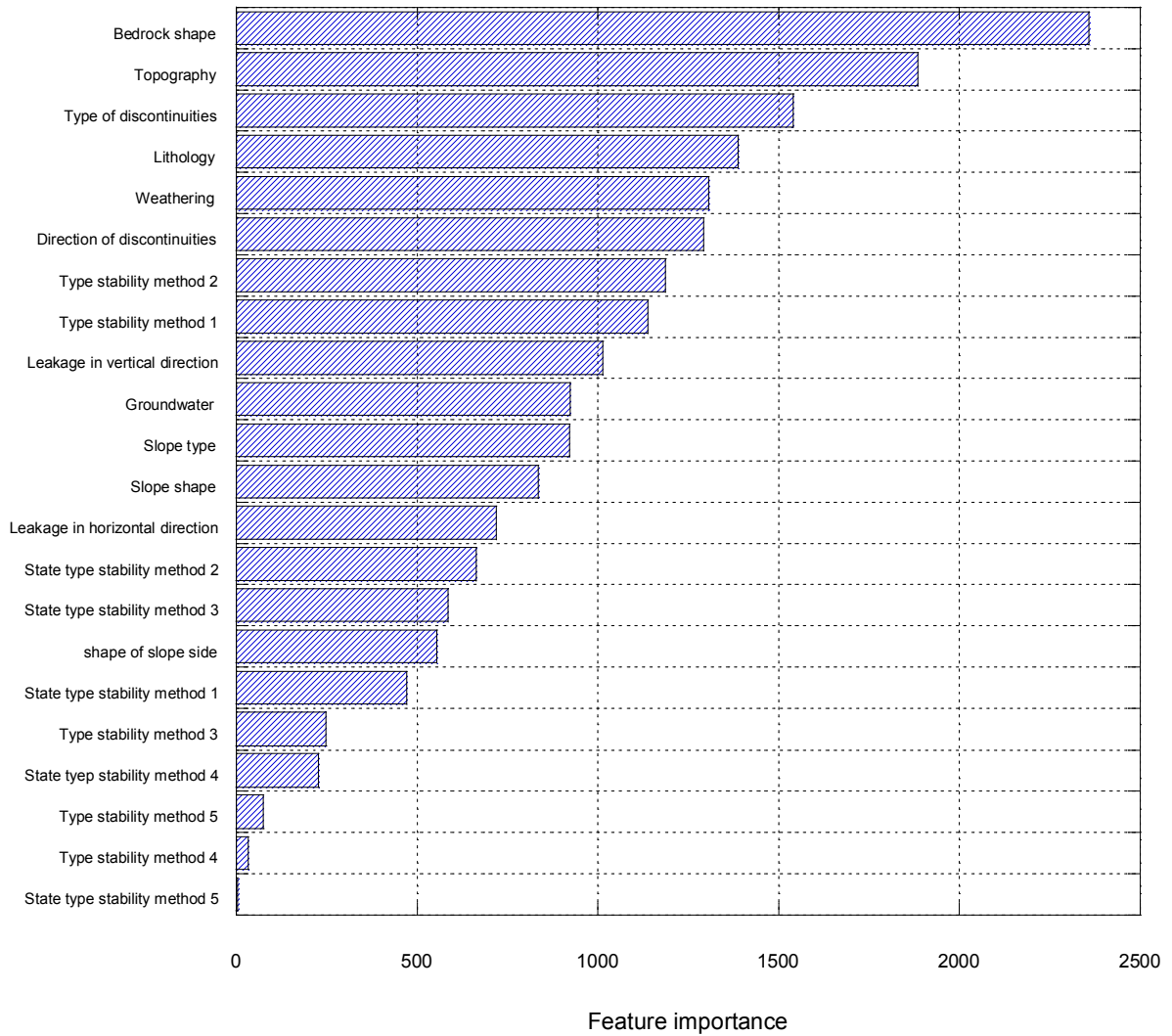


Figure 4-21. Result for feature importance of Case-2 using LightGBM.

4.2.4 LightGBM 비탈면 안정성 예측 모델 : Case-3

수치형과 범주형 데이터를 포함한 31개의 입력변수와 LightGBM을 이용하여 비탈면 안정성 예측 모델에 대해 학습 성능, AUC, 혼동행렬을 사용해 평가하였다. 모델 학습에 대한 학습 성능 평가는 Logloss를 이용하였다. 그 결과에 의하면, Case-2 보다 훈련데이터에 과적합하는 현상이 나타났다(Figure 4-22). 분류 성능 지표로 많이 사용되는 AUC의 경우는 0.783로 산출되었다(Figure 4-23). 훈련 및 시험데이터에 대한 출력데이터의 실제값과 예측값에 대한 빈도는 Figure 4-24과 같이 나타났으며, 이를 통해 산출한 정밀도, 재현율, F1-Score, 정확도의 결과는 다음과 같다(Table 4-8).

훈련데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 정밀도는 85.0%, 안정으로 예측한 경우의 정밀도는 83.0%, 평균 정밀도는 84.0%로 나타났다. 시험데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 정밀도는 80.0%, 안정으로 예측한 경우의 정밀도는 77.0%, 평균 정밀도는 78.5%로 나타났다.

훈련데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 재현율은 83.0%, 안정으로 예측한 경우의 재현율은 86.0%, 평균 재현율은 84.5%로 나타났다. 시험데이터를 사용한 예측 모델에서 비탈면의 상태를 불안정으로 예측한 경우의 재현율은 76.0%, 안정으로 예측한 경우의 재현율은 80.0%, 평균 재현율은 78.0%로 나타났다.

훈련데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 F1-Score는 84.0%, 안정으로 예측한 경우의 F1-Score는 85.0%, 평균 F1-Score는 84.5%로 나타났다. 시험데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 F1-Score는 78.0%, 안정으로 예측한 경우의 F1-Score는 79.0%, 평균 F1-Score는 78.5%로 나타났다.

트리를 분할할 때 사용되는 변수 중요도는 ‘토층심도’, ‘사면상부경사’, ‘사면경사’ 등 ‘소단 개소’와 ‘계곡부 개소’를 제외한 수치형 변수들이 다른 변수들보다 비해 상대적으로 큰 비중을 차지하는 것으로 나타났다(Figure 4-25).

Table 4-8. Performance of Case-3 as a result of classification using LightGBM

		Precision(%)	Recall(%)	F1-Score(%)	Accuracy(%)
Prediction performance for train data	Unstable (Label : 0)	85.0	83.0	84.0	84.0
	Stable (Label : 1)	83.0	86.0	85.0	
	Average	84.0	84.5	84.5	
Prediction performance for test data	Unstable (Label : 0)	80.0	76.0	78.0	78.0
	Stable (Label : 1)	77.0	80.0	79.0	
	Average	78.5	78.0	78.5	

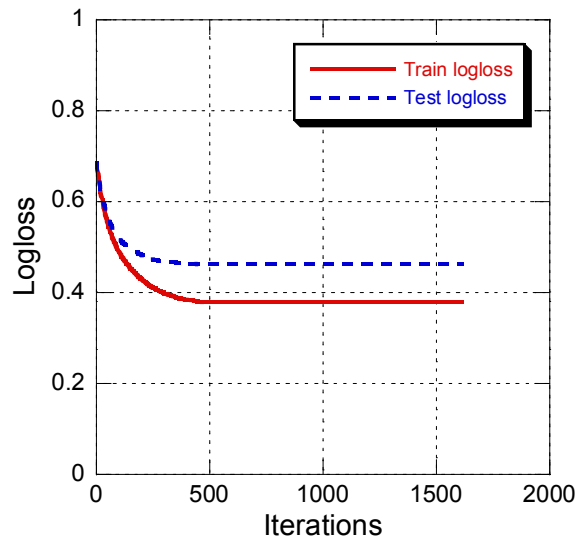


Figure 4-22. Result for prediction performance of Case-3 using LightGBM.

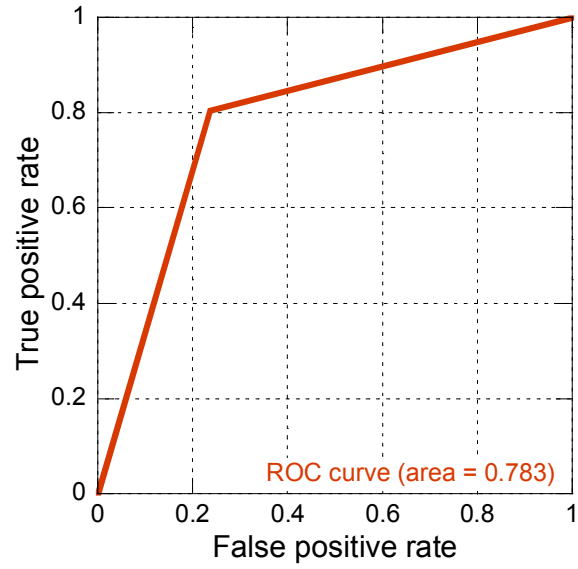


Figure 4-23. Result for AUC of Case-3 using LightGBM.

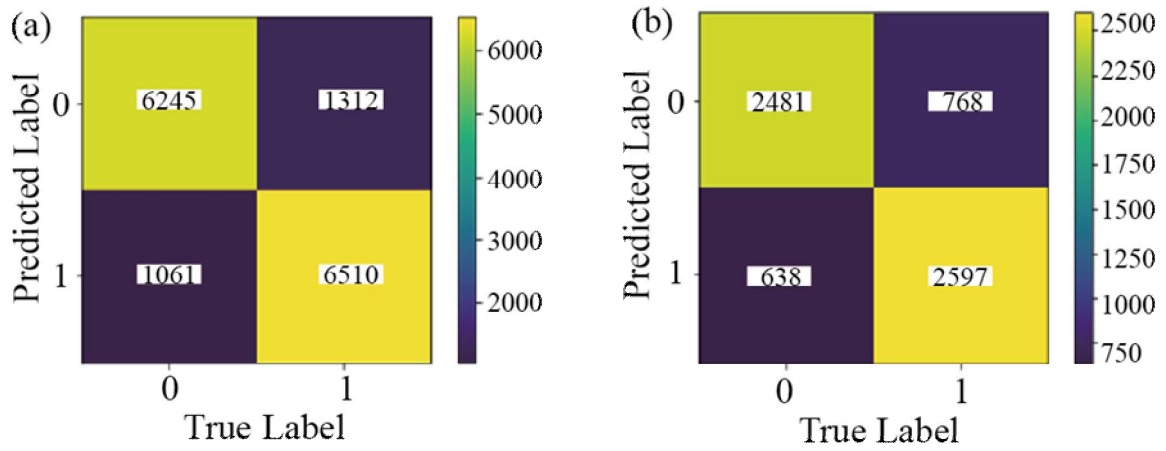


Figure 4-24. Results for confusion matrix of Case-3 on (a) train data and (b) test data using LightGBM.

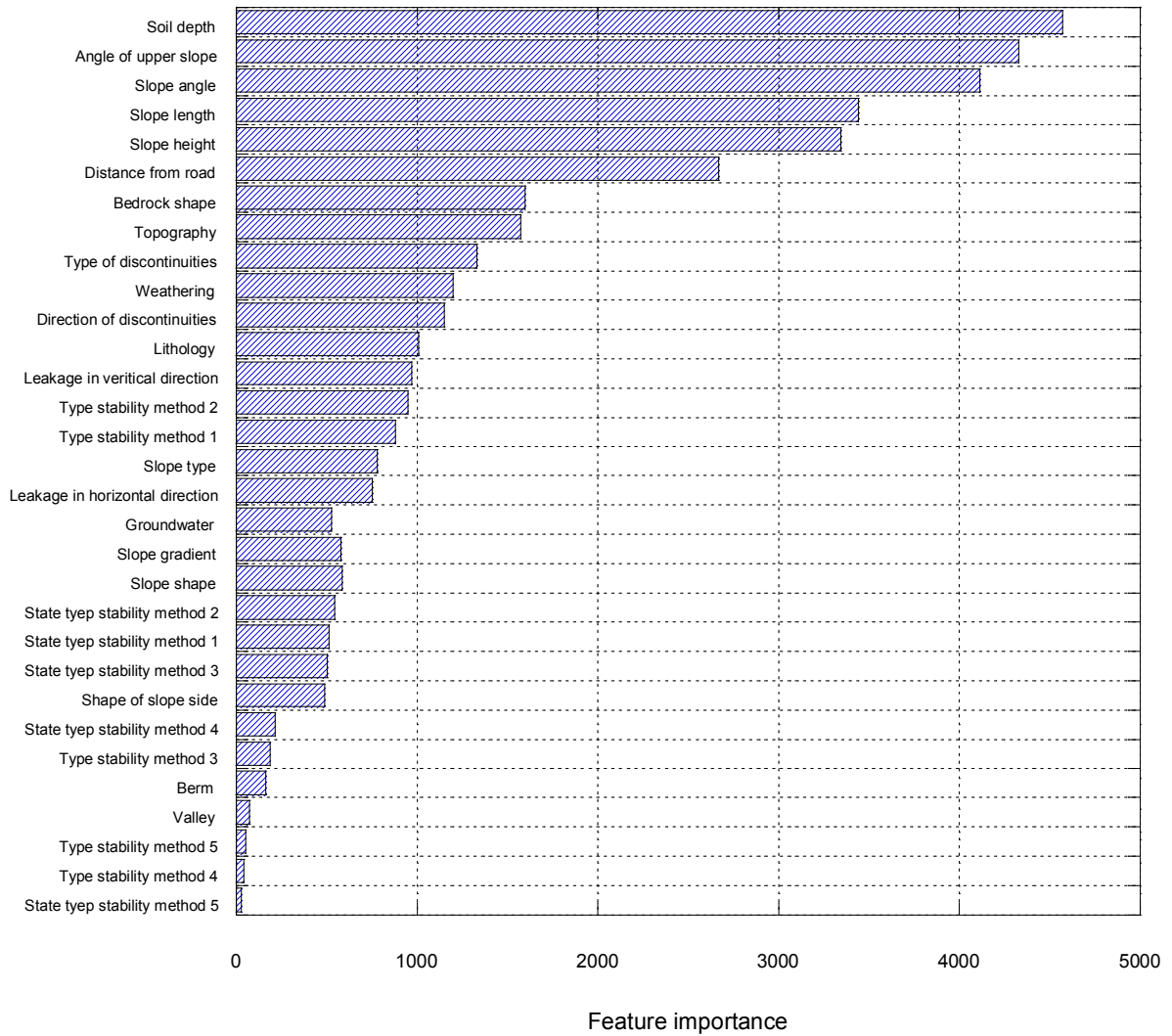


Figure 4-25. Result for feature importance of Case-3 using LightGBM.

4.2.5 LightGBM 비탈면 안정성 예측 모델 결과

비탈면 안정성 예측 모델을 제안하기 위해 LightGBM과 입력데이터에 따른 3가지의 해석 단계를 이용하여 각 모델의 결과를 분석하였다. 비탈면 안정성에 대한 각 변수의 영향을 비교 분석하기 위해 해석 단계를 수치형 데이터 9개로 구성된 Case-1, 범주형 데이터 22개로 구성된 Case-2, 수치형 데이터와 범주형 데이터 31개로 구성된 Case-3으로 구분하였다(Table 3-6). 훈련데이터와 시험데이터 비율은 7:3으로 설정하였고, 초매개변수 최적화는 베이지안 최적화 방법을 사용하였다. 훈련데이터에 대한 과적합을 방지하기 위하여 계층별 k-fold 교차 검증을 5분할 하여 실시하였다. 각 해석 단계에 따른 초매개변수 산정 결과는 Table 4-5과 같다. 모델 학습에 대한 평가는 Logloss를 이용하였고, 모델의 예측 성능 평가는 AUC, 정밀도, 재현율, F1-Score, 정확도를 사용하였다.

3가지 해석 단계에 대한 학습 성능 분석 결과, Case-3이 가장 좋은 성능을 보였고, 다음으로 Case-2, Case-1로 나타났다. Case-1과 Case-2는 훈련데이터와 시험데이터가 유사한 학습 곡선을 나타냈지만, Case-3은 훈련데이터에 약한 과적합 현상을 보였다. AUC는 Case-1의 경우 0.681, Case-2의 경우 0.766, Case-3의 경우 0.783으로 나타났다. 3가지의 해석 단계 중 AUC는 Case-3가 성능이 가장 좋게 나타났고, 그다음 Case-2, Case-1 순으로 나타났다.

Case-1의 예측 성능 결과, 훈련데이터와 시험데이터 사이의 정밀도, 재현율, F1-Score, 그리고 정확도는 약 5%의 차이를 보였다. 출력데이터의 값에 따른 정밀도, 재현율, F1-Score는 훈련데이터와 시험데이터에서 2% 이하로 나타나 서로 유사한 것으로 나타났다. 또한 정밀도와 재현율은 훈련데이터와 시험데이터 모두에서 1~2%의 미세한 차이를 나타냈다(Figure 4-26). Case-2의 예측 성능 결과, 훈련데이터와 시험데이터 사이의 정밀도, 재현율, F1-Score, 그리고 정확도는 약 2%의 차이를 보였다. 출력데이터의 값에 따른 정밀도, 재현율, F1-Score는 훈련데이터와 시험데이터에서 1~4%의 차이를 나타냈다. 또한 정밀도와 재현율은 훈련데이터와 시험데이터에서 2~4%의 차이가 나타났다(Figure 4-26). Case-3의 예측 성능 결과, 훈련데이터와 시험데이터 사이의 정밀도, 재현율, F1-Score, 그리고 정확도는 약 6%의 차이를 보였다. 출력데이터의 값에 따른 정밀도, 재현율, F1-Score는 훈련데이터와 시험데이터에서 1~4%의 차이를 나타냈다. 또한 정밀도와 재현율은 훈련

데이터와 시험데이터에서 2~4%의 차이가 나타났다(Figure 4-26). 예측 성능의 결과를 종합하면 해석 단계에 따른 예측 성능은 Case-3가 가장 좋게 나타났고, 그다음 Case-2, Case-1 순으로 높게 나타났다. LightGBM은 XGBoost와 다르게 정밀도와 재현율의 차이가 4% 이하로 발생하였기 때문에, 각 해석 단계에 따른 예측 성능 불균형은 발생하지 않았다.

Case-1의 변수 중요도는 ‘소단 개소’와 ‘계곡부 개소’를 제외한 ‘사면길이’, ‘사면상부경사’, ‘사면경사’ 등이 다른 변수들에 비해 크게 나타났다. Case-2의 변수 중요도는 ‘암반형태’, ‘주변지형’이 다른 변수들보다 상대적으로 크게 나타났다. Case-3의 변수 중요도는 ‘토층심도’, ‘사면상부경사’, ‘사면경사’, ‘사면길이’, ‘사면높이’, ‘이격거리’ 등의 변수가 다른 변수들에 비해 높게 나타났다. LightGBM을 활용한 비탈면 안정성 예측 모델의 변수 중요도는 전반적으로 XGBoost와 유사하게 범주형 데이터보다 수치형 데이터가 상대적으로 높게 나타났다. 수치형 데이터 중에서 ‘토층심도’, ‘사면상부경사’, ‘사면각도’의 변수 중요도가 높게 나타났고, 범주형 데이터 중에서는 ‘암반형태’와 ‘주변지형’의 변수 중요도가 높게 나타났다.

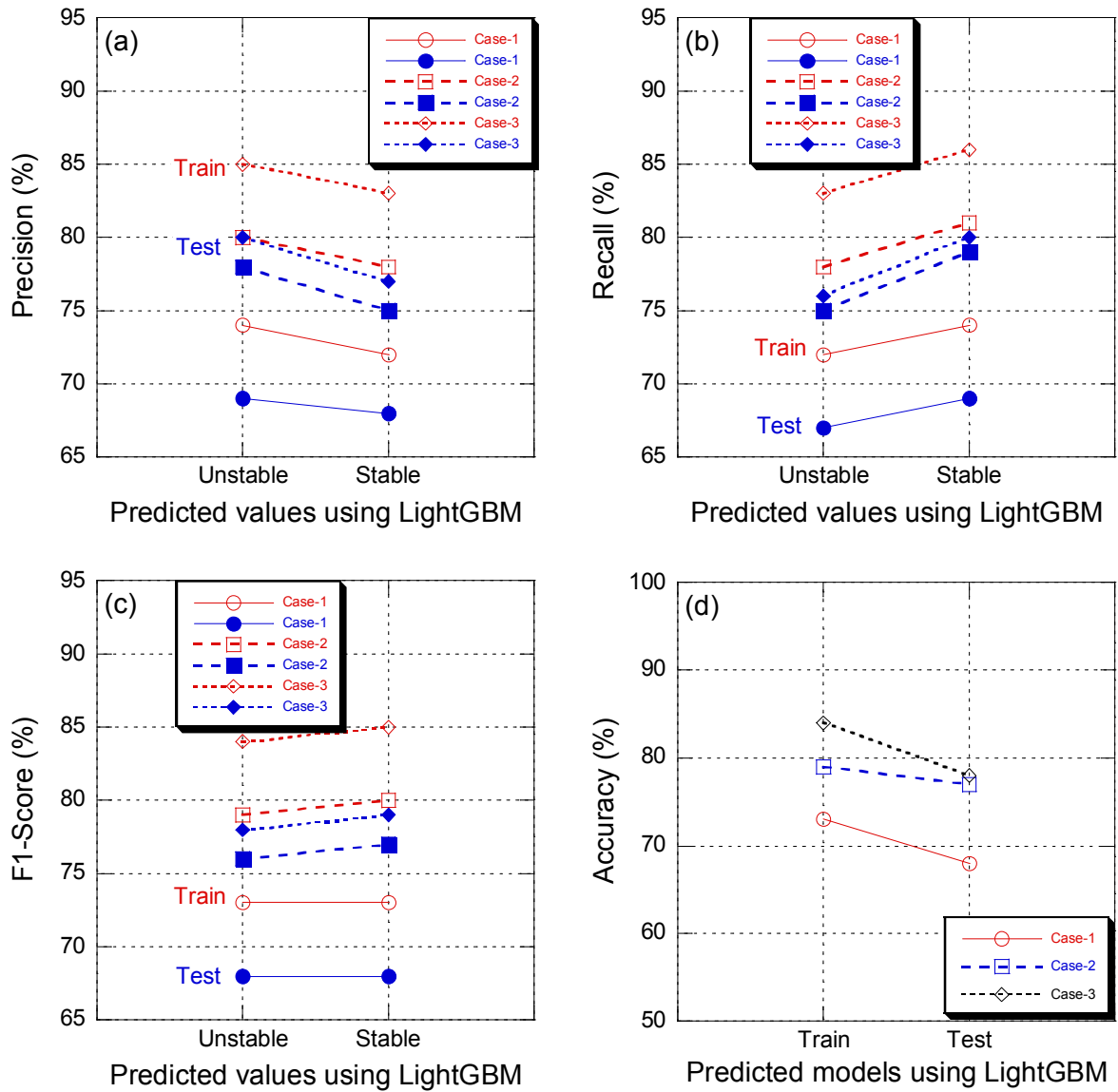


Figure 4-26. Result for slope stability prediction model using LightGBM (a) precision, (b) recall (c) F1-Score, (d) accuracy.

4.3 CatBoost 예측 모델

4.3.1 CatBoost 비탈면 안정성 예측 모델 초매개변수

CatBoost을 이용한 비탈면 안정성 예측 모델을 평가하기 위하여 훈련데이터와 시험데이터 비율은 7:3으로 설정하였다. 초매개변수 최적화는 2.1.3절에서 설명한 베이지안 최적화 방법을 사용하였으며, 훈련데이터에 대한 과적합을 방지하기 위해 계층별 k-fold 교차 검증은 5개 분할하여 실시하였다. 그 결과 Case-2와 Case-3은 Case-1과 다르게 동일한 초매개변수 값이 산출되었다(Table 4-9).

Table 4-9. Results of hyperparameter tuning for CatBoost according to each analysis step

	Case-1	Case-2	Case-3
bagging_temperature	834.01310	628.22796	628.2279665
border_count	69.37383	200.36720	200.36720
boosting_type	Ordered	Ordered	Ordered
depth	5.191896	14.52670	14.52670
eval_metric	Accuracy	Accuracy	Accuracy
iterations	3580.34668	4148.65475	4148.65475
l2_leaf_reg	2.13075	2.7142200	2.7142200
learning_rate	0.009284	0.02145	0.02145
loss_function	Logloss	Logloss	Logloss
min_data_in_leaf	12.65645	8.42800	8.42800
rsm	0.74195	0.20198	0.20198
sampling_frequency	PerTreeLevel	PerTreeLevel	PerTreeLevel
subsample	0.972413	0.98132	0.98132

4.3.2 CatBoost 비탈면 안정성 예측 모델 : Case-1

수치형 데이터 9개와 CatBoost을 이용한 비탈면 안정성 예측 모델에 대해 학습 성능, AUC, 혼동행렬을 사용해 평가하였다. 모델 학습에 대한 학습 성능 평가는 Logloss를 이용하였다. 모델 성능은 학습데이터에서 과적합되지 않고, 학습데이터와 시험데이터에서 유사한 성능을 보였다(Figure 4-27). 분류 성능 지표로 많이 사용되는 AUC의 경우는 0.672로 산출되었다(Figure 4-28). 훈련 및 시험데이터에 대한 출력데이터의 실제값과 예측값에 대한 빈도는 Figure 4-29과 같이 나타났으며, 이를 통해 산출한 정밀도, 재현율, F1-Score, 정확도의 결과는 다음과 같다(Table 4-10).

훈련데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 정밀도는 70.0%, 안정으로 예측한 경우의 정밀도는 69.0%, 평균 정밀도는 69.5%로 나타났다. 시험데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 정밀도는 67.0%, 안정으로 예측한 경우의 정밀도는 66.0%, 평균 정밀도는 66.5%로 나타났다.

훈련데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 재현율은 68.0%, 안정으로 예측한 경우의 재현율은 71.0%, 평균 재현율은 69.5%로 나타났다. 시험데이터를 사용한 예측 모델에서 비탈면의 상태를 불안정으로 예측한 경우의 재현율은 64.0%, 안정으로 예측한 경우의 재현율은 69.0%, 평균 재현율은 66.5%로 나타났다.

훈련데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 F1-Score는 69.0%, 안정으로 예측한 경우의 F1-Score는 70.0%, 평균 F1-Score는 69.5%로 나타났다. 시험데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 F1-Score는 66.0%, 안정으로 예측한 경우의 F1-Score는 67.0%, 평균 F1-Score는 66.5%로 나타났다.

트리를 분할할 때 사용되는 변수 중요도는 ‘토층심도’, ‘사면상부경사’, ‘사면높이’ 순으로 나타났다(Figure 4-30).

Table 4-10. Performance of Case-1 as a result of classification using CatBoost

		Precision	Recall	F1-Score	Accuracy
Prediction performance for train data	Unstable (Label : 0)	70.0	68.0	69.0	70.0
	Stable (Label : 1)	69.0	71.0	70.0	
	Average	69.5	69.5	69.5	
Prediction performance for test data	Unstable (Label : 0)	67.0	64.0	66.0	66.0
	Stable (Label : 1)	66.0	69.0	67.0	
	Average	66.5	66.5	66.5	

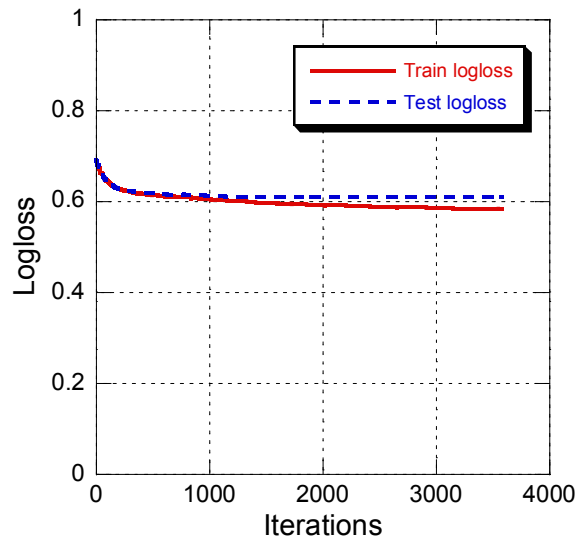


Figure 4-27. Result for prediction performance of Case-1 using CatBoost.

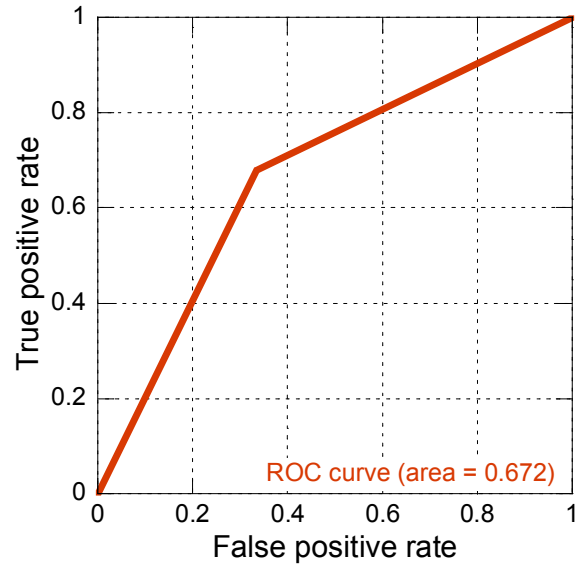


Figure 4-28. Result for AUC of Case-1 using CatBoost.

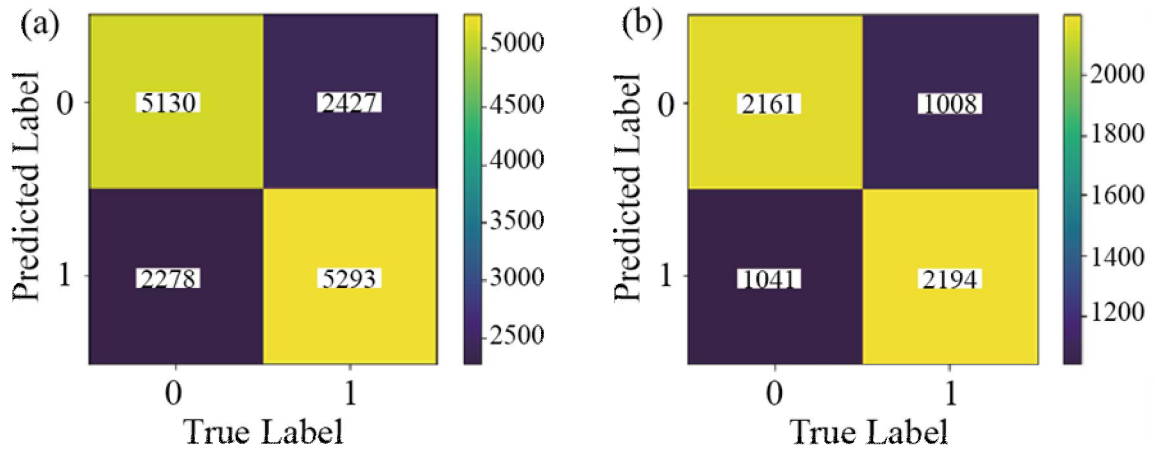


Figure 4-29. Results for confusion matrix of Case-1 on (a) train data and (b) test data using CatBoost.

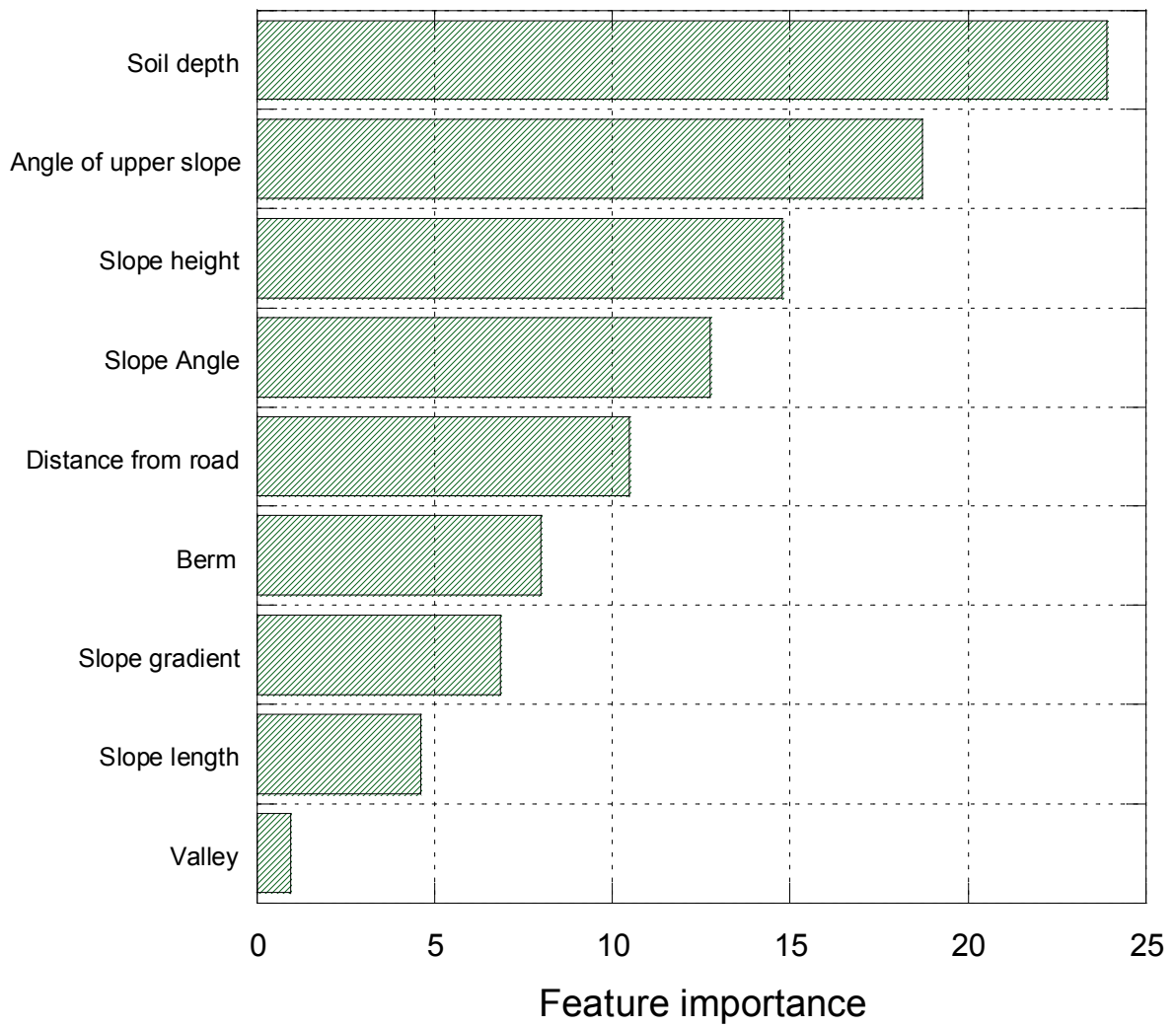


Figure 4-30. Result for feature importance of Case-1 using CatBoost.

4.3.3 CatBoost 비탈면 안정성 예측 모델 : Case-2

범주형 데이터 22개와 CatBoost을 이용한 비탈면 안정성 예측 모델에 대해 학습 성능, AUC, 혼동행렬을 사용해 평가하였다. 모델 학습에 대한 학습 성능 평가는 Logloss를 이용하였고, 그 결과에 의하면 훈련데이터에서 과적합된 현상을 보이는 것으로 나타났다(Figure 4-31). 분류 성능 지표로 많이 사용되는 ROC의 경우는 0.777로 산출되었다(Figure 4-32). 훈련 및 시험데이터에 대한 출력데이터의 실제값과 예측값에 대한 빈도는 Figure 4-33과 같이 나타났으며, 이를 통해 산출한 정밀도, 재현율, F1-Score, 정확도의 결과는 다음과 같다(Table 4-11).

훈련데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 정밀도는 89.0%, 안정으로 예측한 경우의 정밀도는 84.0%, 평균 정밀도는 86.5%로 나타났다. 시험데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 정밀도는 79.0%, 안정으로 예측한 경우의 정밀도는 76.0%, 평균 정밀도는 77.5%로 나타났다.

훈련데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 재현율은 83.0%, 안정으로 예측한 경우의 재현율은 89.0%, 평균 재현율은 86.0%로 나타났다. 시험데이터를 사용한 예측 모델에서 비탈면의 상태를 불안정으로 예측한 경우의 재현율은 75.0%, 안정으로 예측한 경우의 재현율은 80.0%, 평균 재현율은 77.5%로 나타났다.

훈련데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 F1-Score는 86.0%, 안정으로 예측한 경우의 F1-Score는 87.0%, 평균 F1-Score는 86.5%로 나타났다. 시험데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 F1-Score는 77.0%, 안정으로 예측한 경우의 F1-Score는 78.0%, 평균 F1-Score는 77.5%로 나타났다.

트리를 분할할 때 변수 중요도는 ‘불연속면 방향성’, ‘측면형상’, ‘주변지형’, ‘불연속면 종류’ 순으로 나타났다(Figure 4-34).

Table 4-11. Performance of Case-2 as a result of classification using CatBoost

		Precision	Recall	F1-Score	Accuracy
Prediction performance for train data	Unstable (Label : 0)	89.0	83.0	86.0	86.0
	Stable (Label : 1)	84.0	89.0	87.0	
	Average	86.5	86.0	86.5	
Prediction performance for test data	Unstable (Label : 0)	79.0	75.0	77.0	78.0
	Stable (Label : 1)	76.0	80.0	78.0	
	Average	77.5	77.5	77.5	

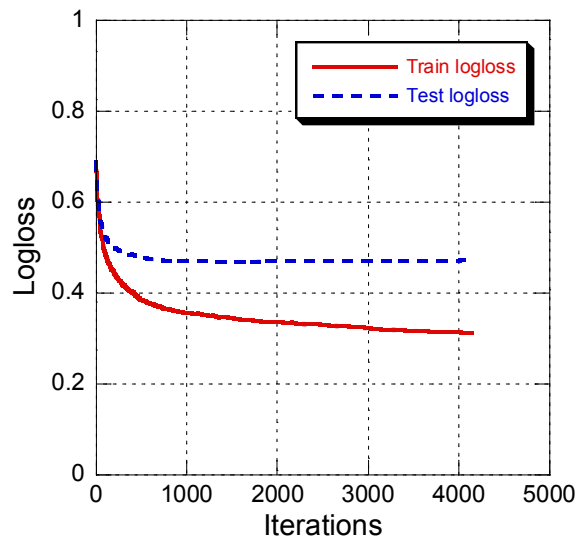


Figure 4-31. Result for prediction performance of Case-2 using CatBoost.

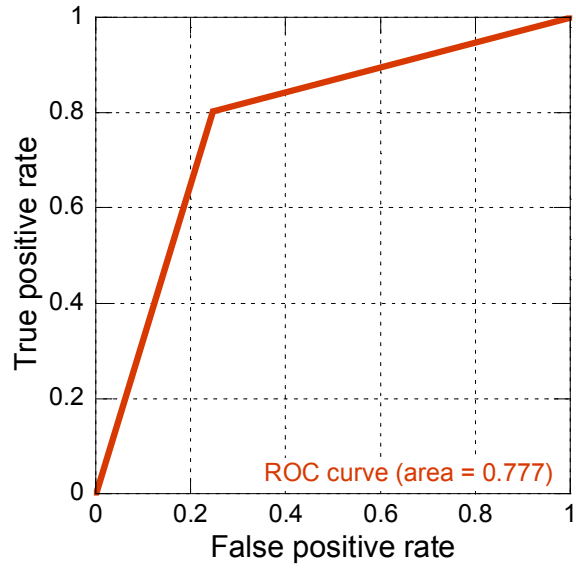


Figure 4-32. Result for AUC of Case-2 using CatBoost.

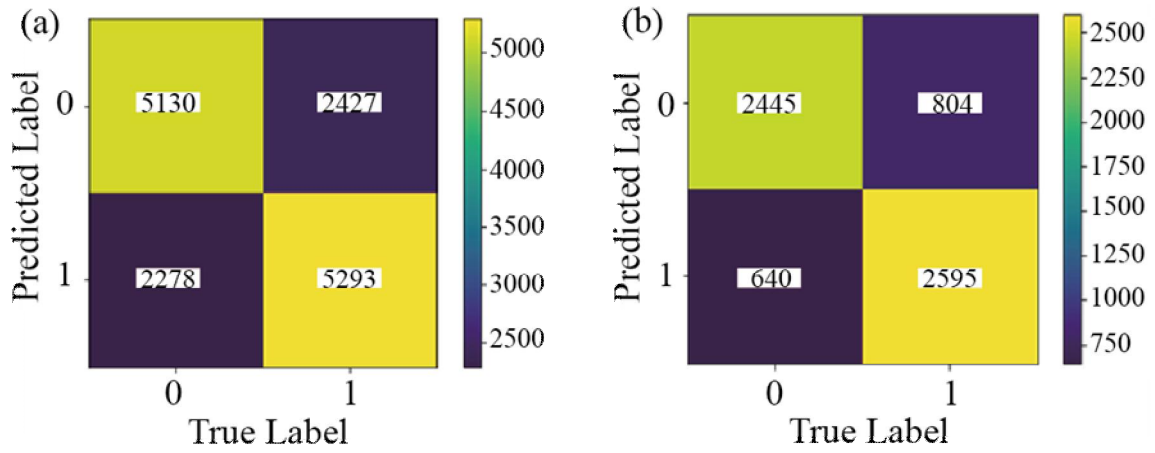


Figure 4-33. Results for confusion matrix of Case-2 on (a) train data and (b) test data using CatBoost.

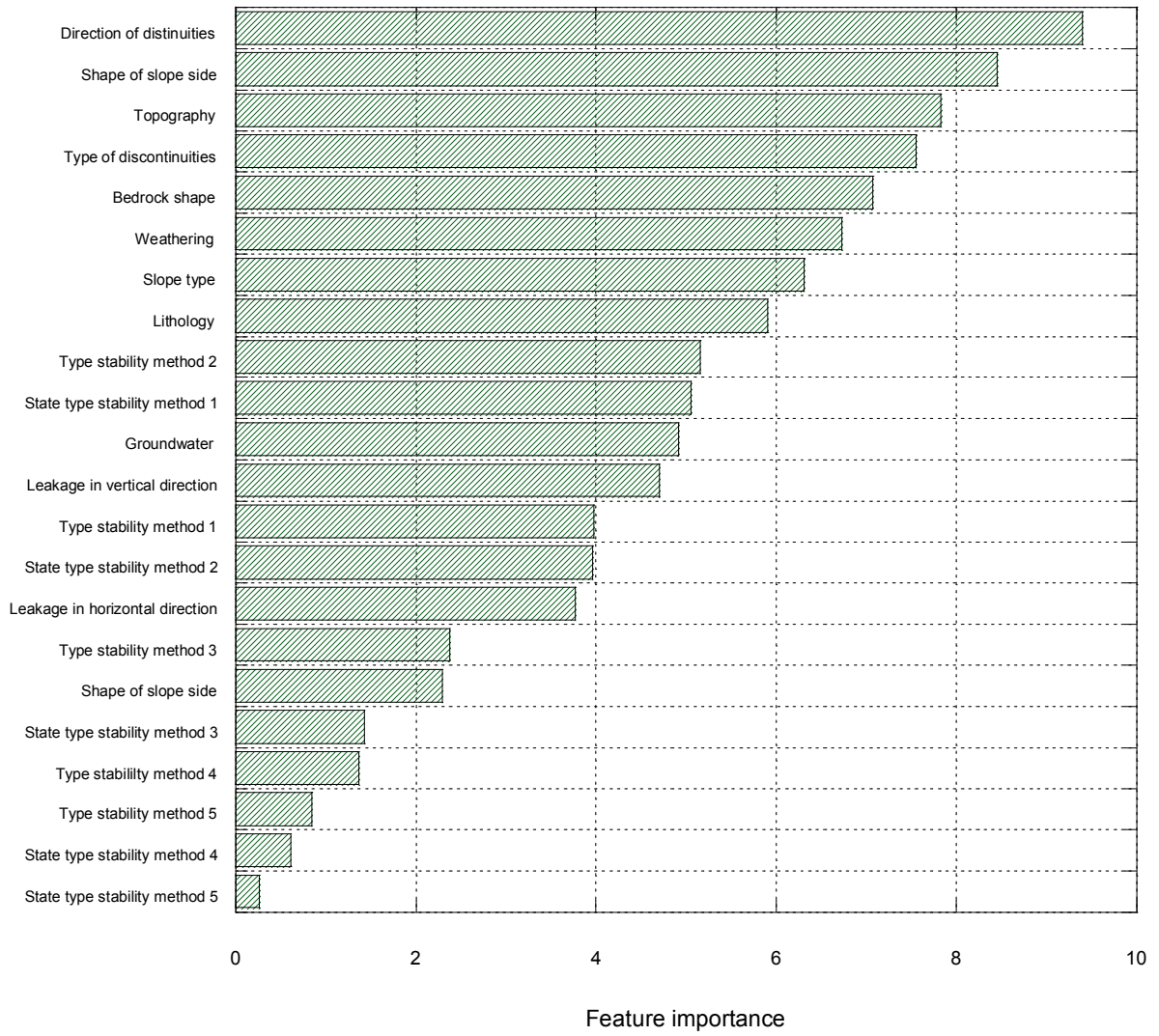


Figure 4-34. Result for feature importance of Case-2 using CatBoost.

4.3.4 CatBoost 비탈면 안정성 예측 모델 : Case-3

수치형과 범주형 데이터를 포함한 31개의 입력변수와 CatBoost을 이용한 비탈면 안정성 예측 모델에 대해 학습 성능, AUC, 혼동행렬 사용해 평가하였다. 모델 학습에 대한 학습 성능 평가는 Logloss를 이용하였으며 그 결과에 의하면, 훈련데이터에서 과적합된 현상을 보이는 것으로 나타났다(Figure 4-35). 분류 성능 지표로 많이 사용되는 ROC의 경우는 0.794로 산출되었다(Figure 4-36). 훈련 및 시험데이터에 대한 출력데이터의 실제값과 예측값에 대한 빈도는 Figure 4-37과 같이 나타났으며, 이를 통해 산출한 정밀도, 재현율, F1-Score, 정확도의 결과는 다음과 같다(Table 4-12).

훈련데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 정밀도는 92.0%, 안정으로 예측한 경우의 정밀도는 88.0%, 평균 정밀도는 90.0%로 나타났다. 시험데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 정밀도는 80.0%, 안정으로 예측한 경우의 정밀도는 78.0%, 평균 정밀도는 79.0%로 나타났다.

훈련데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 재현율은 87.0%, 안정으로 예측한 경우의 재현율은 93.0%, 평균 재현율은 90.0%로 나타났다. 시험데이터를 사용한 예측 모델에서 비탈면의 상태를 불안정으로 예측한 경우의 재현율은 78.0%, 안정으로 예측한 경우의 재현율은 81.0%, 평균 재현율은 79.5%로 나타났다.

훈련데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 F1-Score는 90.0%, 안정으로 예측한 경우의 F1-Score는 90.0%, 평균 F1-Score는 90.0%로 나타났다. 시험데이터를 사용한 예측 모델에서 비탈면 상태를 불안정으로 예측한 경우의 F1-Score는 79.0%, 안정으로 예측한 경우의 F1-Score는 80.0%, 평균 F1-Score는 79.5%로 나타났다.

트리를 분할할 때 변수의 중요도는 ‘토층심도’가 가장 크게 나타났으며, ‘불연속면 방향성’, ‘측면형상’, ‘불연속면 종류’, ‘주변지형’ 순으로 XGBoost, LightGBM과 다르게 범주형 변수가 더 높은 중요도를 갖는 것으로 나타났다(Figure 4-38).

Table 4-12. Performance of Case-3 as a result of classification using CatBoost

		Precision	Recall	F1-Score	Accuracy
Prediction performance for train data	Unstable (Label : 0)	92.0	87.0	90.0	90.0
	Stable (Label : 1)	88.0	93.0	90.0	
	Average	90.0	90.0	90.0	
Prediction performance for test data	Unstable (Label : 0)	80.0	78.0	79.0	79.0
	Stable (Label : 1)	78.0	81.0	80.0	
	Average	79.0	79.5	79.5	

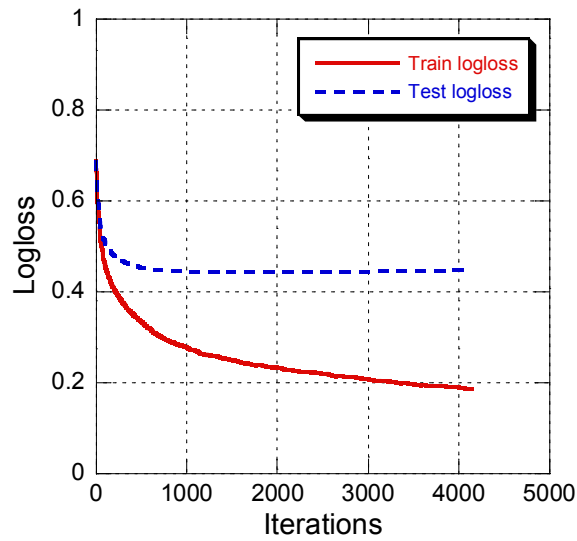


Figure 4-35. Result for prediction performance of Case-3 using CatBoost.

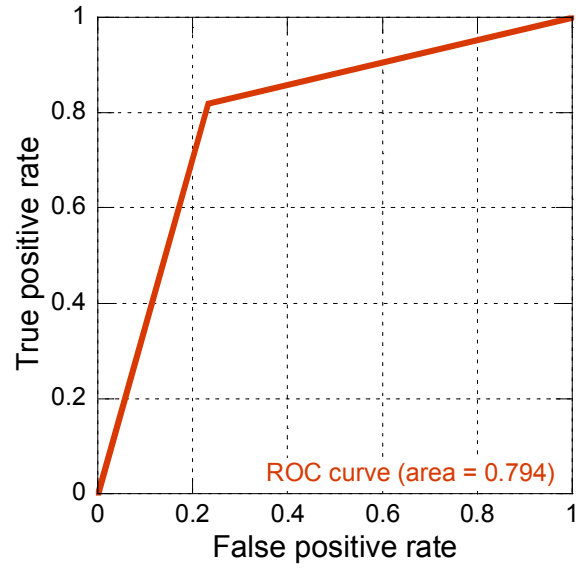


Figure 4-36. Result for AUC of Case-3 using CatBoost.

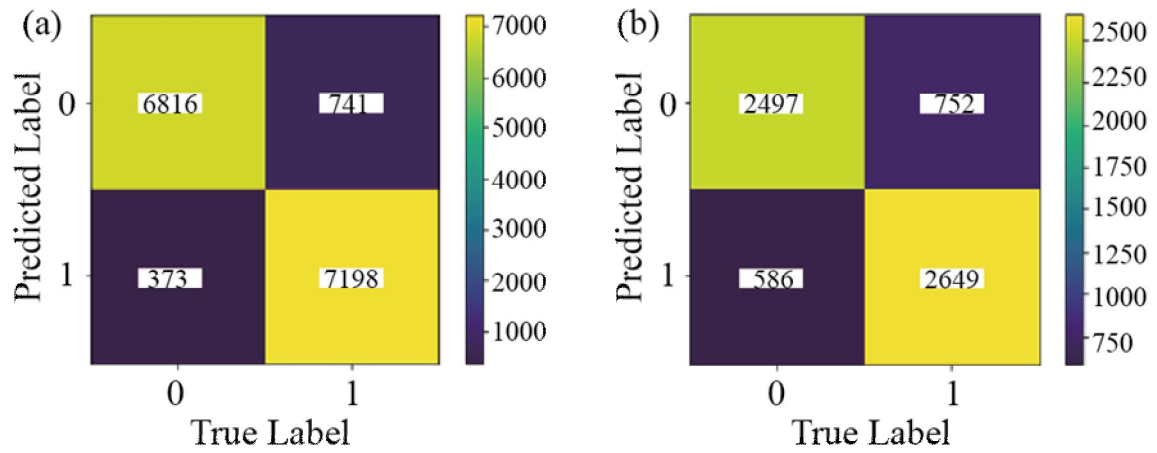


Figure 4-37. Results for confusion matrix of Case-3 on (a) train data and (b) test data using CatBoost.

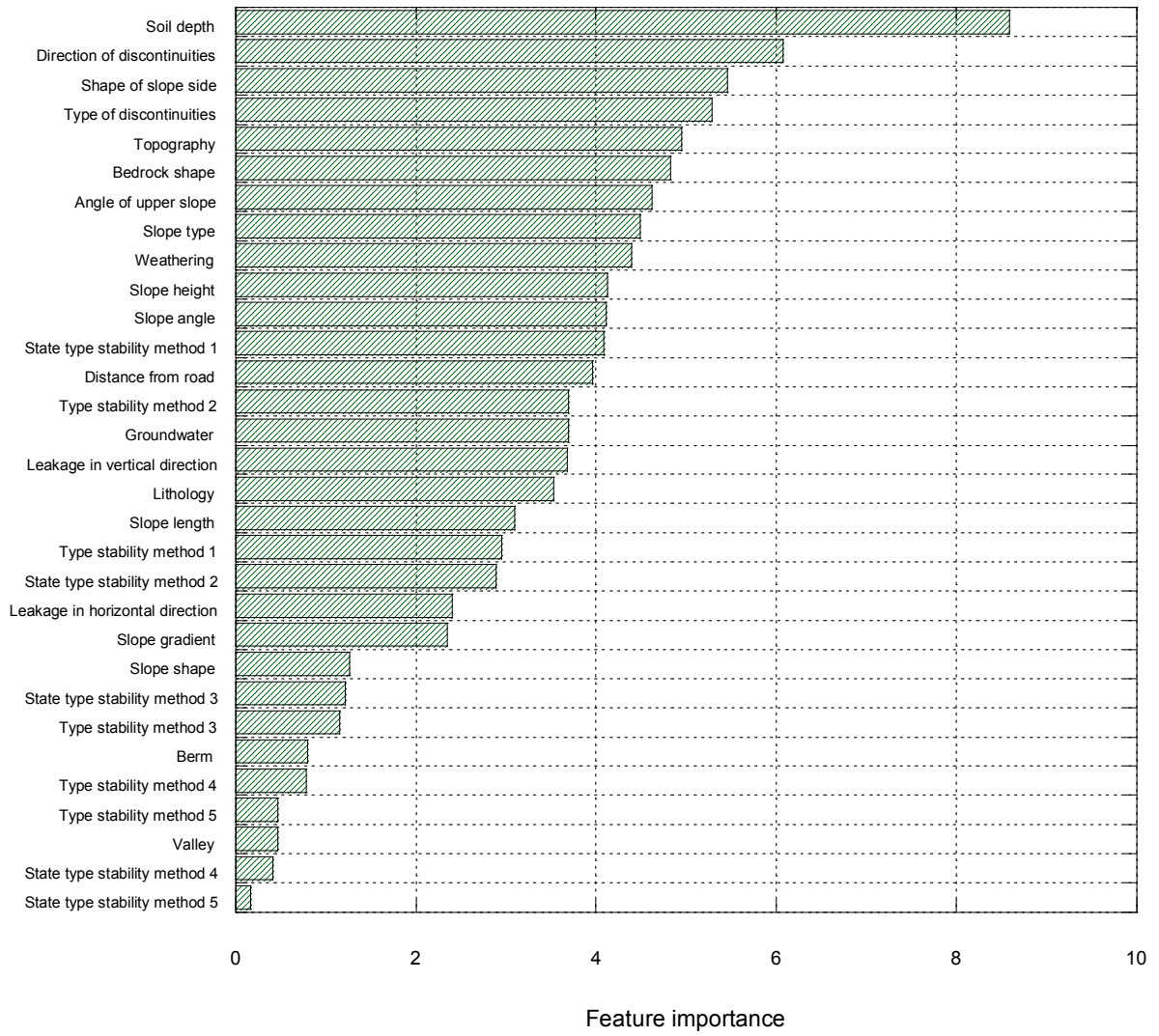


Figure 4-38. Result for feature importance of Case-3 using CatBoost.

4.3.5 CatBoost 비탈면 안정성 예측 모델 결과

비탈면 안정성 예측 모델을 제안하기 위해 CatBoost와 입력데이터에 따른 3가지의 해석 단계를 이용하여 각 모델의 결과를 분석하였다. 비탈면 안정성에 대한 각 변수의 영향을 비교 분석하기 위해 해석 단계를 수치형 데이터 9개로 구성된 Case-1, 범주형 데이터 22개로 구성된 Case-2, 수치형 데이터와 범주형 데이터 31개로 구성된 Case-3으로 구분하였다(Table 3-6). 훈련데이터와 시험데이터 비율은 7:3으로 설정하였고, 초매개변수 최적화는 베이지안 최적화 방법을 사용하였다. 훈련데이터에 대한 과적합을 방지하기 위하여 계층별 k-fold 교차 검증을 5분할 하여 실시하였다. 각 해석 단계에 따른 초매개변수 산정 결과는 Table 4-9과 같다. 모델 학습에 대한 평가는 Logloss를 이용하였고, 모델의 예측 성능 평가는 AUC, 정밀도, 재현율, F1-Score, 정확도를 사용하였다.

3가지 해석 단계에 대한 학습 성능 분석 결과, Case-3이 가장 좋은 성능을 보였고, 다음으로 Case-2, Case-1로 나타났다. Case-1은 훈련데이터와 시험데이터가 유사한 학습 곡선을 나타냈지만, Case-2와 Case-3은 훈련데이터에 과적합 현상을 보였다. AUC는 Case-1의 경우 0.672, Case-2의 경우 0.777, Case-3의 경우 0.794으로 나타났다. 3가지의 해석 단계 중 AUC는 Case-3가 성능이 가장 좋게 나타났고, 그다음 Case-2, Case-1 순으로 나타났다.

Case-1의 예측 성능 결과, 훈련데이터와 시험데이터 사이의 정밀도, 재현율, F1-Score, 그리고 정확도는 3~4% 차이를 나타냈고, 출력데이터의 값에 따른 정밀도, 재현율, F1-Score는 훈련데이터와 시험데이터 모두에서 1~5%의 차이를 나타냈다. 또한 정밀도와 재현율은 훈련데이터와 시험데이터에서 2~3% 미세한 차이가 나타났다(Figure 4-39). Case-2의 예측 성능 결과, 훈련데이터와 시험데이터 사이의 정밀도, 재현율, F1-Score, 그리고 정확도는 약 8~9%의 차이를 보였고, 출력데이터의 값에 따른 정밀도, 재현율, F1-Score는 훈련데이터와 시험데이터에서 1~6%의 차이를 나타냈다. 또한 정밀도와 재현율은 훈련데이터와 시험데이터에서 3~6%의 차이가 나타났다(Figure 4-39). Case-3의 예측 성능 결과, 훈련데이터와 시험데이터 사이의 정밀도, 재현율, F1-Score, 그리고 정확도는 약 11%의 차이를 보였고, 출력데이터의 값에 따른 정밀도, 재현율, F1-Score는 훈련데이터와 시험데이터 모두에서 1~6%의 차이가 나타났다. 또한 정밀도와 재현율은 훈련데이터와 시

험데이터에서 2~5%의 차이를 나타냈다(Figure 4-39). 예측 성능의 결과를 종합하면 해석 단계에 따른 예측 성능은 Case-3가 가장 좋게 나타났고, 그다음 Case-2, Case-1 순으로 높게 나타났다. CatBoost도 LightGBM과 유사하게 정밀도와 재현율의 차이가 약 5% 이하로 발생했기 때문에, 각 해석 단계에 따른 예측 성능 불균형 현상이 발생하지 않았다. 하지만, 다른 모델들과 다르게 입력데이터 수가 많아지고, 입력데이터 형태가 범주형일 경우 훈련데이터에 심하게 과적합 되는 현상이 나타났다.

Case-1의 변수 중요도는 ‘토층심도’가 가장 높게 나타났고, 그다음 ‘사면상부경사’, ‘사면높이’ 순으로 나타났다. Case-2의 변수 중요도는 ‘불연속면 방향성’, ‘측면형상’, ‘주변지형’ 등의 변수가 순서대로 높게 나타났다. Case-3의 변수 중요도는 다른 머신러닝 모듈의 Case-3와 다르게 ‘토층심도’가 가장 크게 나타났고, 그다음 범주형 데이터인 ‘불연속면 방향성’, ‘측면형상’, ‘불연속면 종류’ 등의 변수들이 순서대로 높게 나타났다. 변수 중요도는 다른 머신러닝 모델과 다르게 수치형 데이터보다 범주형 데이터가 상대적으로 높게 나타났고, 범주형 데이터 중에서 ‘불연속면 방향성’, ‘측면형상’, ‘주변지형’의 변수 중요도가 높게 나타났고, 수치형 데이터 중에서는 ‘토층심도’의 변수 중요도가 높게 나타났다.

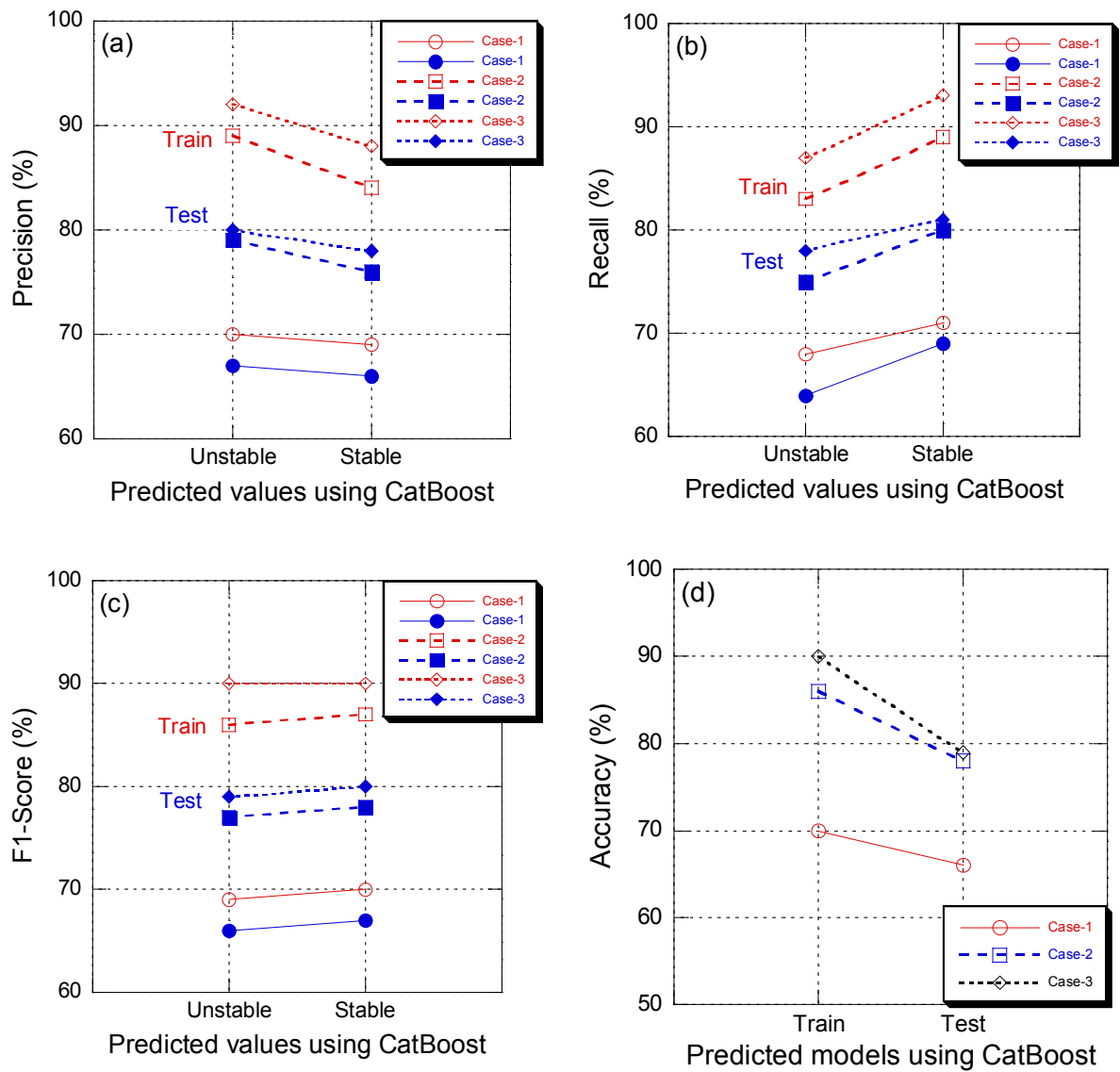


Figure 4-39. Result for slope stability prediction model using CatBoost (a) precision, (b) recall (c) F1-Score, (d) accuracy.

5. 토의

5.1 머신러닝 방법에 따른 비탈면 안정성 예측 모델 결과

비탈면 안정성 예측 모델을 제안하기 위해 XGBoost, LightGBM, 그리고 CatBoost와 비탈면 조사자료를 이용해 예측 모델을 비교 분석하였다. 이를 위해 먼저 비탈면 조사자료를 파악하여 입력데이터를 선정한 후, 데이터 전처리를 실시하였다. 비탈면 조사자료는 크게 객관적 자료에 해당하는 기초조사자료와 전문가에 의해 조사된 주관적 조사자료인 정밀조사자료로 구분할 수 있다. 이들 중 관리번호, 위치정보, 향후 비탈면 보강 계획 등의 변수를 제외한 50개의 변수를 선정하였고, 각 데이터 형태에 따라 이상치 제거, 구간화 등의 데이터 전처리를 실시하였다. 또한, 통계적 검증을 실시하여 입력데이터 31개와 출력데이터 1개를 선정하였다. 비탈면 안정성에 영향을 미치는 입력데이터의 변수 중요도와 모델을 교차검정하기 위하여 해석 단계를 Case-1, Case-2, Case-3로 분류하였다. Case-1은 9개의 수치형 데이터로만 구성되었고, Case-2는 22개의 범주형 자료로 구성되었으며, Case-3는 수치형과 범주형 데이터 31개로 구성되었다. 각 해석 단계에 대하여 학습 성과 예측 성능 그리고 변수 중요도를 산정하였다. 모델 학습에 사용한 훈련데이터와 시험데이터의 비율은 7:3으로 설정하였고, 초매개변수 최적화는 베이지안 최적화 방법을 사용하였다. 훈련데이터에 대한 과적합을 방지하기 위하여 계층별 k-fold 교차 검증을 5분할 하여 실시하였다.

(1) 머신러닝 방법에 따른 Case-1의 비탈면 안정성 예측 모델의 예측 성능 평가

수치형 데이터 9개로 이루어진 Case-1과 XGBoost, LightGBM, 그리고 CatBoost를 이용해 비탈면 안정성 예측 모델 평가를 수행한 결과는 다음과 같다 (Figure 5-1, Table 5-1).

Case-1의 훈련데이터에서 예측 모델이 비탈면의 상태를 불안정으로 예측한 XGBoost의 정밀도는 70.0%, LightGBM의 정밀도는 74.0%, CatBoost의 정밀도는 70.0%로 나타났고, Case-1의 훈련데이터에서 예측 모델이 비탈면의 상태를 안정으

로 예측한 XGBoost의 정밀도는 66.0%, LightGBM의 정밀도는 72.0%, CatBoost의 정밀도는 69.0%로 나타났다. Case-1의 훈련데이터를 이용한 XGBoost의 평균 정밀도는 68.0%, LightGBM의 평균 정밀도는 73.0%, CatBoost의 평균 정밀도는 69.5%로 산출되었다. Case-1의 시험데이터에서 예측 모델이 비탈면의 상태를 불안정으로 예측한 XGBoost의 정밀도는 69.0%, LightGBM의 정밀도는 69.0%, CatBoost의 정밀도는 67.0%로 나타났고, Case-1의 시험데이터에서 예측 모델이 비탈면의 상태를 안정으로 예측한 XGBoost의 정밀도는 65.0%, LightGBM의 정밀도는 68.0%, CatBoost의 정밀도는 66.0%로 나타났다. Case-1의 시험데이터를 이용한 XGBoost의 평균 정밀도는 67.0%, LightGBM의 평균 정밀도는 68.5%, CatBoost의 평균 정밀도는 66.5%로 산출되었다(Figure 5-1(a)).

Case-1의 훈련데이터에서 예측 모델이 비탈면의 상태를 불안정으로 예측한 XGBoost의 재현율은 61.0%, LightGBM의 재현율은 72.0%, CatBoost의 재현율은 68.0%로 나타났고, Case-1의 훈련데이터에서 예측 모델이 비탈면의 상태를 안정으로 예측한 XGBoost의 재현율은 74.0%, LightGBM의 재현율은 74.0%, CatBoost의 재현율은 71.0%로 나타났다. Case-1의 훈련데이터를 이용한 XGBoost의 평균 재현율은 67.5%, LightGBM의 평균 재현율은 73.0%, CatBoost의 평균 재현율은 69.5%로 산출되었다. Case-1의 시험데이터에서 예측 모델이 비탈면의 상태를 불안정으로 예측한 XGBoost의 재현율은 61.0%, LightGBM의 재현율은 67.0%, CatBoost의 재현율은 64.0%로 나타났고, Case-1의 시험데이터에서 예측 모델이 비탈면의 상태를 안정으로 예측한 XGBoost의 재현율은 73.0%, LightGBM의 재현율은 69.0%, CatBoost의 재현율은 69.0%로 나타났다. Case-1의 시험데이터를 이용한 XGBoost의 평균 재현율은 67.0%, LightGBM의 평균 재현율은 68.0%, CatBoost의 평균 재현율은 66.5%로 산출되었다(Figure 5-1(b)).

Case-1의 훈련데이터에서 예측 모델이 비탈면의 상태를 불안정으로 예측한 XGBoost의 F1-Score는 66.0%, LightGBM의 F1-Score는 73.0%, CatBoost의 F1-Score는 69.0%로 나타났고, Case-1의 훈련데이터에서 예측 모델이 비탈면의 상태를 안정으로 예측한 XGBoost의 F1-Score는 70.0%, LightGBM의 F1-Score는 73.0%, CatBoost의 F1-Score는 70.0%로 나타났다. Case-1의 훈련데이터를 이용한 XGBoost의 평균 F1-Score는 68.0%, LightGBM의 평균 F1-Score는 73.0%, CatBoost의 평균 F1-Score는 69.5%로 산출되었다. Case-1의 시험데이터에서 예측 모델이 비탈면의 상태를 불안정으로 예측한 XGBoost의 F1-Score는 65.0%,

LightGBM의 F1-Score는 68.0%, CatBoost의 F1-Score는 66.0%로 나타났고, Case-1의 시험데이터에서 예측 모델이 비탈면의 상태를 안정으로 예측한 XGBoost의 F1-Score는 67.0%, LightGBM의 F1-Score는 68.0%, CatBoost의 F1-Score는 67.0%로 나타났다. Case-1의 시험데이터를 이용한 XGBoost의 평균 F1-Score는 66.0%, LightGBM의 평균 F1-Score는 68.0%, CatBoost의 평균 F1-Score는 66.5%로 산출되었다(Figure 5-1(c)).

Case-1의 훈련데이터를 이용한 비탈면 안정성 예측 모델의 정확도는 XGBoost의 경우 68.0%, LightGBM 73.0%, CatBoost 70.0%, 시험데이터를 이용한 비탈면 안정성 예측 모델의 정확도는 XGBoost의 경우 67.0%, LightGBM의 경우 68%, CatBoost의 경우 66%로 나타났다(Figure 5-1(d)). Case-1의 AUC는 LightGBM이 0.681로 가장 높게 나타났고, 그다음 CatBoost, XGBoost 순으로 각각 0.672, 0.668로 나타났다.

변수 중요도는 XGBoost의 경우 ‘소단 개소’, ‘계곡부 개소’를 제외한 ‘사면길이’, ‘사면각도’, ‘토층심도’ 등의 변수가 유사하게 나타났다. LightGBM도 역시 ‘소단 개소’, ‘계곡부 개소’를 제외한 변수들이 상대적으로 높은 중요도를 나타냈고, 특히 ‘사면길이’, ‘사면상부경사’가 다른 변수들에 비해 크게 나타났다. CatBoost는 ‘토층심도’가 가장 높게 나타났고, ‘사면상부경사’, ‘사면높이’, ‘사면각도’ 순으로 나타났다. 3개의 모델을 종합하면 ‘소단 개소’, ‘계곡부 개소’를 제외한 변수들의 변수 중요도가 높게 나타났고, 그들 중 ‘토층심도’, ‘사면길이’, ‘사면상부경사’가 다른 변수들에 비해 큰 변수 중요도를 갖는 것으로 판단된다.

전체적으로 XGBoost, LightGBM, CatBoost와 수치형 데이터 9개로 구성된 Case-1을 비탈면 안정성 예측 모델로 분석한 결과 학습 성능에 있어 3개의 모델에서 모두 훈련데이터에 과적합 되지 않고, 양호한 학습 곡선을 나타냈다. 예측 성능 결과에서도 60% 후반에서 70% 초반의 준수한 예측 성능을 나타냈다. 특히, 정밀도와 재현율에 있어, LightGBM과 CatBoost의 경우 각 예측 성능의 결과가 큰 차이를 나타내지 않았지만, XGBoost의 경우 8~9%의 차이가 나타나 예측 성능 불균형이 발생하였다. 수치형 데이터의 경우 데이터 전처리 시 ‘소단 개소’와 ‘계곡부의 개소’를 제외한 7개의 변수에 대해 이상치를 제거하였기 때문에 일반화가 잘되어있다고 판단할 수 있다. 이로 인해 모델 학습과 예측 성능에서 훈련데이터와 시험데이터가 서로 유사한 성능을 나타낸 것으로 사료 된다. 각각의 머신러닝 모델의 목적함수와 트리 분할 방법이 다르기 때문에 XGBoost에서 예측 성능 불균형이 발생한 것으로 판단된다.

Table 5-1. Result for prediction performance of slope stability model using Case-1 depending on machine learning methods

			XGBoost	LightGBM	CatBoost
Precision	Train data	Unstable	70.0	74.0	70.0
		Stable	66.0	72.0	69.0
		Average	68.0	73.0	69.5
	Test data	Unstable	69.0	69.0	67.0
		Stable	65.0	68.0	66.0
		Average	67.0	68.5	66.5
Recall	Train data	Unstable	61.0	72.0	68.0
		Stable	74.0	74.0	71.0
		Average	67.5	73.0	69.5
	Test data	Unstable	61.0	67.0	64.0
		Stable	73.0	69.0	69.0
		Average	67.0	68.0	66.5
F1-Score	Train data	Unstable	66.0	73.0	69.0
		Stable	70.0	73.0	70.0
		Average	68.0	73.0	69.5
	Test data	Unstable	65.0	68.0	66.0
		Stable	67.0	68.0	67.0
		Average	66.0	68.0	66.5
Accuracy	Train data		68.0	73.0	70.0
	Test data		67.0	68.0	66.0

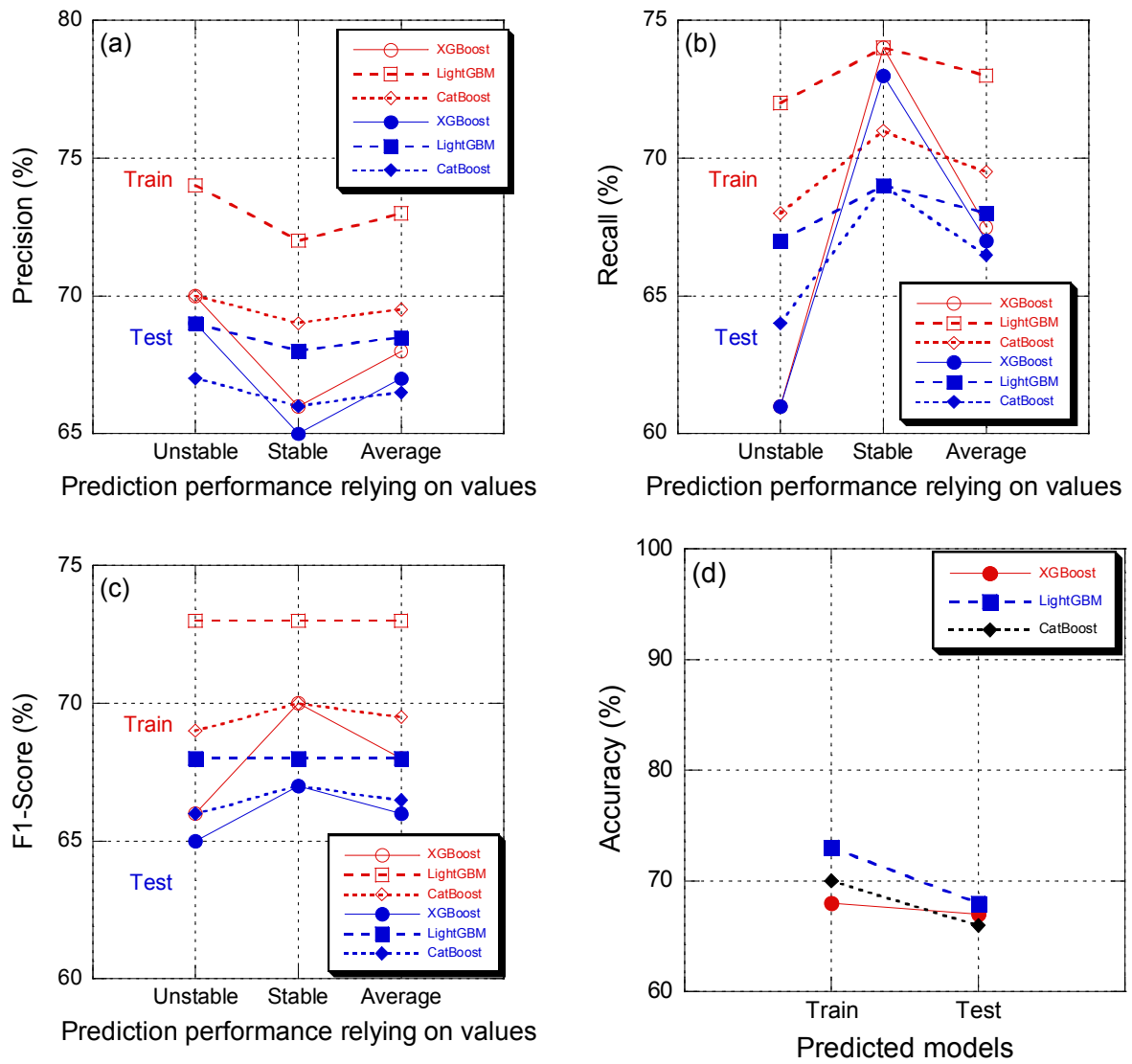


Figure 5-1. Result for prediction performance of slope stability model using Case-1 depending on machine learning methods (a) precision, (b) recall, (c) F1-Score (d) accuracy.

(2) 머신러닝 방법에 따른 Case-2의 비탈면 안정성 예측 모델의 예측 성능 평가

범주형 데이터 22개로 이루어진 Case-2와 XGBoost, LightGBM, 그리고 CatBoost를 이용해 비탈면 안정성 예측 모델 평가를 수행한 결과는 다음과 같다 (Figure 5-2, Table 5-2).

Case-2의 훈련데이터에서 예측 모델이 비탈면의 상태를 불안정으로 예측한 XGBoost의 정밀도는 87.0%, LightGBM의 정밀도는 80.0%, CatBoost의 정밀도는 89.0%로 나타났고, Case-2의 훈련데이터에서 비탈면의 상태를 안정으로 예측한 XGBoost의 정밀도는 79.0%, LightGBM의 정밀도는 78.0%, CatBoost의 정밀도는 84.0%로 나타났다. Case-2의 훈련데이터를 이용한 XGBoost의 평균 정밀도는 83.0%, LightGBM의 평균 정밀도는 79.0%, CatBoost의 평균 정밀도는 86.5%으로 산출되었다. Case-2의 시험데이터에서 예측 모델이 비탈면의 상태를 불안정으로 예측한 XGBoost의 정밀도는 81.0%, LightGBM의 정밀도는 78.0%, CatBoost의 정밀도는 79.0%로 나타났고, Case-2의 시험데이터에서 비탈면의 상태를 안정으로 예측한 XGBoost의 정밀도는 74.0%, LightGBM의 정밀도는 75.0%, CatBoost의 정밀도는 76.0%로 나타났다. Case-2의 시험데이터를 이용한 XGBoost의 평균 정밀도는 77.5%, LightGBM의 평균 정밀도는 76.5%, CatBoost의 평균 정밀도는 77.5%으로 산출되었다(Figure 5-2(a)).

Case-2의 훈련데이터에서 예측 모델이 비탈면의 상태를 불안정으로 예측한 XGBoost의 재현율은 76.0%, LightGBM의 재현율은 78.0%, CatBoost의 재현율은 83.0%로 나타났고, Case-2의 훈련데이터에서 비탈면의 상태를 안정으로 예측한 XGBoost의 재현율은 89.0%, LightGBM의 재현율은 81.0%, CatBoost의 재현율은 89.0%로 나타났다. Case-2의 훈련데이터를 이용한 XGBoost의 평균 재현율은 82.5%, LightGBM의 평균 재현율은 79.5%, CatBoost의 평균 재현율은 86.0%으로 산출되었다. Case-2의 시험데이터에서 예측 모델이 비탈면의 상태를 불안정으로 예측한 XGBoost의 재현율은 70.0%, LightGBM의 재현율은 75.0%, CatBoost의 재현율은 75.0%로 나타났고, Case-2의 시험데이터에서 예측 모델이 비탈면의 상태를 안정으로 예측한 XGBoost의 재현율은 84.0%, LightGBM의 재현율은 79.0%, CatBoost의 재현율은 80.0%로 나타났다. Case-2의 시험데이터를 이용한 XGBoost의 평균 재현율은 77.0%, LightGBM의 평균 재현율은 77.0%, CatBoost의 평균 재현율은 77.5%으로 산출되었다(Figure 5-2(b)).

Case-2의 훈련데이터에서 예측 모델이 비탈면의 상태를 불안정으로 예측한 XGBoost의 F1-Score는 82.0%, LightGBM의 F1-Score는 79.0%, CatBoost의 F1-Score는 86.0%로 나타났고, Case-2의 훈련데이터에서 예측 모델이 비탈면의 상태를 안정으로 예측한 XGBoost의 F1-Score는 84.0%, LightGBM의 F1-Score는 80.0%, CatBoost의 F1-Score는 87.0%로 나타났다. Case-2의 훈련데이터를 이용한 XGBoost의 평균 F1-Score는 83.0%, LightGBM의 평균 F1-Score는 79.5%, CatBoost의 평균 F1-Score는 86.5%으로 산출되었다. Case-2의 시험데이터에서 예측 모델이 비탈면의 상태를 불안정으로 예측한 XGBoost의 F1-Score는 75.0%, LightGBM의 F1-Score는 76.0%, CatBoost의 F1-Score는 77.0%로 나타났고, Case-2의 시험데이터에서 예측 모델이 비탈면의 상태를 안정으로 예측한 XGBoost의 F1-Score는 79.0%, LightGBM의 F1-Score는 77.0%, CatBoost의 F1-Score는 78.0%로 나타났다. Case-2의 시험데이터를 이용한 XGBoost의 평균 F1-Score는 77.0%, LightGBM의 평균 F1-Score는 76.5%, CatBoost의 평균 F1-Score는 77.5%으로 산출되었다(Figure 5-2(c)).

Case-2의 훈련데이터를 이용한 비탈면 안정성 예측 모델의 정확도는 XGBoost의 경우 83.0%, LightGBM 79.0%, CatBoost 86.0%, Case-2의 시험데이터를 이용한 비탈면 안정성 예측 모델의 정확도는 XGBoost의 경우 77.0%, LightGBM의 경우 77%, CatBoost의 경우 78%로 나타났다(Figure 5-2(d)). Case-2의 AUC는 CatBoost가 0.777로 가장 높게 나타났고, 그다음 XGBoost, LightGBM 순으로 각각 0.771, 0.766으로 나타났다.

변수 중요도는 XGBoost의 경우 ‘주변지형’, ‘암종’, ‘암반형태’, ‘비탈면 보강공법 2번의 공법 종류’, ‘풍화도’가 다른 변수들에 비해 상대적으로 크게 나타났다. LightGBM의 경우에는 ‘암반형태’, ‘주변지형’이 다른 변수들에 비해 크게 나타났고, CatBoost의 경우 ‘불연속면 방향성’, ‘측면형상’, ‘주변지형’ 순으로 나타났다. 3개의 모델을 종합하면 범주형 데이터 22개 중에서 ‘암반형태’, ‘주변지형’, ‘불연속면 방향성’이 다른 변수들에 비해 상대적으로 큰 변수 중요도를 갖는 것으로 판단된다.

전체적으로 XGBoost, LightGBM, CatBoost와 범주형 데이터 22개로 구성된 Case-2에 대해 비탈면 안정성 예측 모델로 분석한 결과 학습 성능은 CatBoost가 가장 크고, 그다음 XGBoost, LightGBM 순으로 나타났다. 하지만, CatBoost는 학습과정에 있어 훈련데이터에 과적합하는 현상이 나타났다. 예측 성능은 70% 중반에서 80% 후반의 준수한 예측 성능을 나타냈다. 특히, 정밀도와 재현율에 있어

Case-1과 동일하게 LightGBM과 CatBoost의 경우 각 예측 성능의 결과에 큰 차이가 나타나지 않았지만, XGBoost의 경우 10~11%의 차이가 나타나 예측 성능 불균형이 Case-1 보다 더 크게 발생하였다. 이러한 현상을 보이는 요인으로는 범주형 데이터의 특성과 지질학적 및 지형학적 특성상 어느 한 값에 편향되었거나, 각 머신러닝 모델의 목적함수와 트리 분할 방법이 다르기 때문에 약간의 차이가 발생한 것으로 판단된다.

Table 5-2. Result for prediction performance of slope stability model using Case-2 depending on machine learning methods

			XGBoost	LightGBM	CatBoost
Precision	Train data	Unstable	87.0	80.0	89.0
		Stable	79.0	78.0	84.0
		Average	83.0	79.0	86.5
	Test data	Unstable	81.0	78.0	79.0
		Stable	74.0	75.0	76.0
		Average	77.5	76.5	77.5
Recall	Train data	Unstable	76.0	78.0	83.0
		Stable	89.0	81.0	89.0
		Average	82.5	79.5	86.0
	Test data	Unstable	70.0	75.0	75.0
		Stable	84.0	79.0	80.0
		Average	77.0	77.0	77.5
F1-Score	Train data	Unstable	82.0	79.0	86.0
		Stable	84.0	80.0	87.0
		Average	83.0	79.5	86.5
	Test data	Unstable	75.0	76.0	77.0
		Stable	79.0	77.0	78.0
		Average	77.0	76.5	77.5
Accuracy	Train data		83.0	79.0	86.0
	Test data		77.0	77.0	78.0

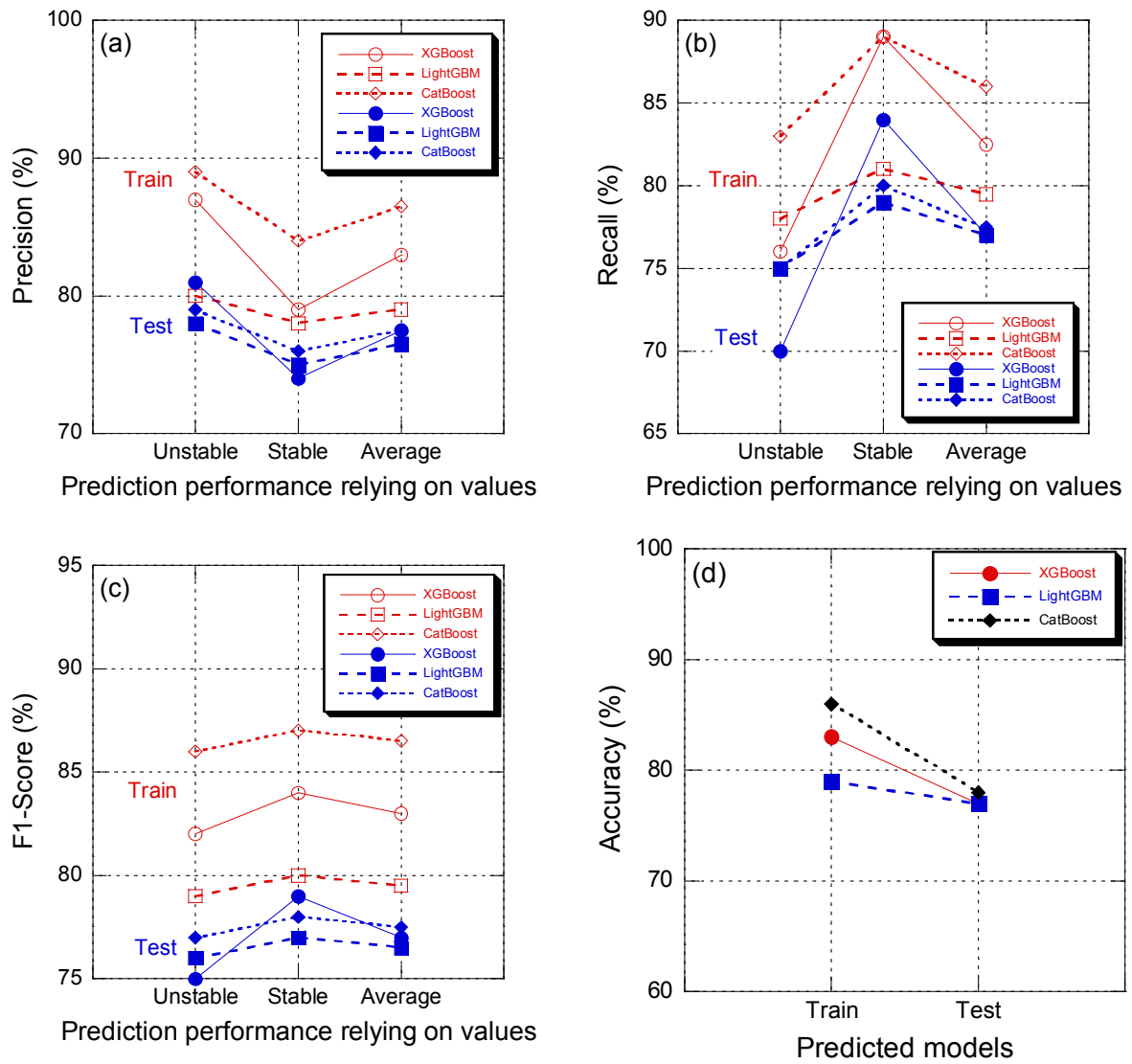


Figure 5-2. Result for prediction performance of slope stability model using Case-2 depending on machine learning methods (a) precision, (b) recall, (c) F1-Score (d) accuracy.

(3) 머신러닝 방법에 따른 Case-3의 비탈면 안정성 예측 모델의 예측 성능 평가

수치형 데이터 9개와 범주형 데이터 22개로 이루어진 Case-3와 XGBoost, LightGBM, 그리고 CatBoost를 이용해 비탈면 안정성 예측 모델 평가를 수행한 결과는 다음과 같다(Figure 5-3, Table 5-3).

Case-3의 훈련데이터에서 예측 모델이 비탈면의 상태를 불안정으로 예측한 XGBoost의 정밀도는 89.0%, LightGBM의 정밀도는 85.0%, CatBoost의 정밀도는 92.0%로 나타났고, Case-3의 훈련데이터에서 예측 모델이 비탈면의 상태를 안정으로 예측한 XGBoost의 정밀도는 76.0%, LightGBM의 정밀도는 83.0%, CatBoost의 정밀도는 88.0%로 나타났다. Case-1의 훈련데이터를 이용한 XGBoost의 평균 정밀도는 82.5%, LightGBM의 평균 정밀도는 84.0%, CatBoost의 평균 정밀도는 90.0%로 산출되었다. Case-3의 시험데이터에서 예측 모델이 비탈면의 상태를 불안정으로 예측한 XGBoost의 정밀도는 85.0%, LightGBM의 정밀도는 80.0%, CatBoost의 정밀도는 80.0%로 나타났고, Case-3의 시험데이터에서 예측 모델이 비탈면의 상태를 안정으로 예측한 XGBoost의 정밀도는 72.0%, LightGBM의 정밀도는 77.0%, CatBoost의 정밀도는 78.0%로 나타났다. Case-1의 시험데이터를 이용한 XGBoost의 평균 정밀도는 78.5%, LightGBM의 평균 정밀도는 78.5%, CatBoost의 평균 정밀도는 79.0%로 산출되었다(Figure 5-3(a)).

Case-3의 훈련데이터에서 예측 모델이 비탈면의 상태를 불안정으로 예측한 XGBoost의 재현율은 71.0%, LightGBM의 재현율은 83.0%, CatBoost의 재현율은 87.0%로 나타났고, Case-3의 훈련데이터에서 예측 모델이 비탈면의 상태를 안정으로 예측한 XGBoost의 재현율은 92.0%, LightGBM의 재현율은 86.0%, CatBoost의 재현율은 93.0%로 나타났다. Case-3의 훈련데이터를 이용한 XGBoost의 평균 재현율은 81.5%, LightGBM의 평균 재현율은 84.5%, CatBoost의 평균 재현율은 90.0%로 산출되었다. Case-3의 시험데이터에서 예측 모델이 비탈면의 상태를 불안정으로 예측한 XGBoost의 재현율은 66.0%, LightGBM의 재현율은 76.0%, CatBoost의 재현율은 78.0%로 나타났고, Case-3의 시험데이터에서 예측 모델이 비탈면의 상태를 안정으로 예측한 XGBoost의 재현율은 88.0%, LightGBM의 재현율은 80.0%, CatBoost의 재현율은 81.0%로 나타났다. Case-3의 시험데이터를 이용한 XGBoost의 평균 재현율은 77.0%, LightGBM의 평균 재현율은 78.0%, CatBoost의 평균 재현율은 79.5%로 산출되었다(Figure 5-3(b)).

Case-3의 훈련데이터에서 예측 모델이 비탈면의 상태를 불안정으로 예측한 XGBoost의 F1-Score는 79.0%, LightGBM의 F1-Score는 84.0%, CatBoost의 F1-Score는 90.0%로 나타났고, Case-3의 훈련데이터에서 예측 모델이 비탈면의 상태를 안정으로 예측한 XGBoost의 F1-Score는 83.0%, LightGBM의 F1-Score는 85.0%, CatBoost의 F1-Score는 90.0%로 나타났다. Case-1의 훈련데이터를 이용한 XGBoost의 평균 F1-Score는 81.0%, LightGBM의 평균 F1-Score는 84.5%, CatBoost의 평균 F1-Score는 90.0%으로 산출되었다. Case-3의 시험데이터에서 예측 모델이 비탈면의 상태를 불안정으로 예측한 XGBoost의 F1-Score는 74.0%, LightGBM의 F1-Score는 78.0%, CatBoost의 F1-Score는 79.0%로 나타났고, Case-3의 시험데이터에서 예측 모델이 비탈면의 상태를 안정으로 예측한 XGBoost의 F1-Score는 79.0%, LightGBM의 F1-Score는 79.0%, CatBoost의 F1-Score는 79.5%로 나타났다. Case-3의 시험데이터를 이용한 XGBoost의 평균 F1-Score는 76.5%, LightGBM의 평균 F1-Score는 78.5%, CatBoost의 평균 F1-Score는 79.5%으로 산출되었다(Figure 5-3(c)).

Case-3의 훈련데이터를 이용한 비탈면 안정성 예측 모델의 정확도는 XGBoost의 경우 81.0%, LightGBM 84.0%, CatBoost 90.0%, Case-3의 시험데이터를 이용한 비탈면 안정성 예측 모델의 정확도는 XGBoost의 경우 77.0%, LightGBM의 경우 78%, CatBoost의 경우 79%로 나타났다(Figure 5-3(d)). Case-3의 AUC는 CatBoost가 0.794로 가장 높게 나타났고, 그 다음 LightGBM, XGBoost 순으로 각각 0.783, 0.777로 나타났다.

변수 중요도는 XGBoost의 경우 전반적으로 범주형 데이터보다 수치형 데이터의 변수 중요도가 높게 나타났고, 특히 ‘사면길이’, ‘사면상부경사’, ‘토층심도’ 등의 변수가 다른 변수들에 비해 크게 나타났다. LightGBM의 경우 XGBoost와 유사하게 범주형 데이터보다 수치형 데이터가 상대적으로 높게 나타났고, 그들 중에서도 ‘토층심도’, ‘사면상부경사’, ‘사면경사’, ‘사면길이’ 등의 변수가 다른 변수들에 비해 높게 나타났다. CatBoost는 다른 모델들과 다르게 전반적으로 수치형 데이터보다 범주형 데이터의 변수 중요도가 상대적으로 높게 나타났다. ‘토층심도’가 가장 크게 나타났고, 그다음 ‘불연속면 방향성’, ‘측면형상’, ‘불연속면 종류’ 등의 변수들이 높게 나타났다. 3개의 모델을 종합하면 ‘토층심도’와 ‘사면경사’의 변수가 다른 변수들에 비해 변수 중요도가 높게 나타났다.

전체적으로 XGBoost, LightGBM, CatBoost와 수치형 및 범주형 데이터 31개로

구성된 Case-3에 대한 비탈면 안정성 예측 모델을 분석한 결과, 학습 성능은 CatBoost가 가장 크고, 그다음 LightGBM, XGBoost 순으로 나타났다. 하지만, CatBoost는 학습과정에 있어 훈련데이터에 과적합 현상이 나타났고, Case-2 보다 더 크게 발생하였다. 예측 성능은 70 후반에서 90%까지로, 해석 단계 중 가장 높게 나타났다. 특히, 정밀도와 재현율에 있어 이전 해석 단계들과 동일하게 LightGBM 과 CatBoost의 경우 각 예측 성능의 결과에 큰 차이가 나타나지 않았지만, XGBoost의 경우 약 16~19%의 차이가 나타나 예측 성능 불균형이 Case-3에서 가장 크게 발생하였다. 이러한 현상을 보이는 요인으로는 수치형 데이터와 비교했을 때, 범주형 데이터의 특성과 지질학적 및 지형학적 특성상 어느 한 값에 편향되었거나, 각 머신러닝 모델의 목적함수와 트리 분할 방법이 다르기 때문에 약간의 차이가 발생한 것으로 판단된다.

Table 5-3. Result for prediction performance of slope stability model using Case-3 depending on machine learning methods

			XGBoost	LightGBM	CatBoost
Precision	Train data	Unstable	89.0	85.0	92.0
		Stable	76.0	83.0	88.0
		Average	82.5	84.0	90.0
	Test data	Unstable	85.0	80.0	80.0
		Stable	72.0	77.0	78.0
		Average	78.5	78.5	79.0
Recall	Train data	Unstable	71.0	83.0	87.0
		Stable	92.0	86.0	93.0
		Average	81.5	84.5	90.0
	Test data	Unstable	66.0	76.0	78.0
		Stable	88.0	80.0	81.0
		Average	77.0	78.0	79.5
F1-Score	Train data	Unstable	79.0	84.0	90.0
		Stable	83.0	85.0	90.0
		Average	81.0	84.5	90.0
	Test data	Unstable	74.0	78.0	79.0
		Stable	79.0	79.0	80.0
		Average	76.5	78.5	79.5
Accuracy	Train data		81.0	84.0	90.0
	Test data		77.0	78.0	79.0

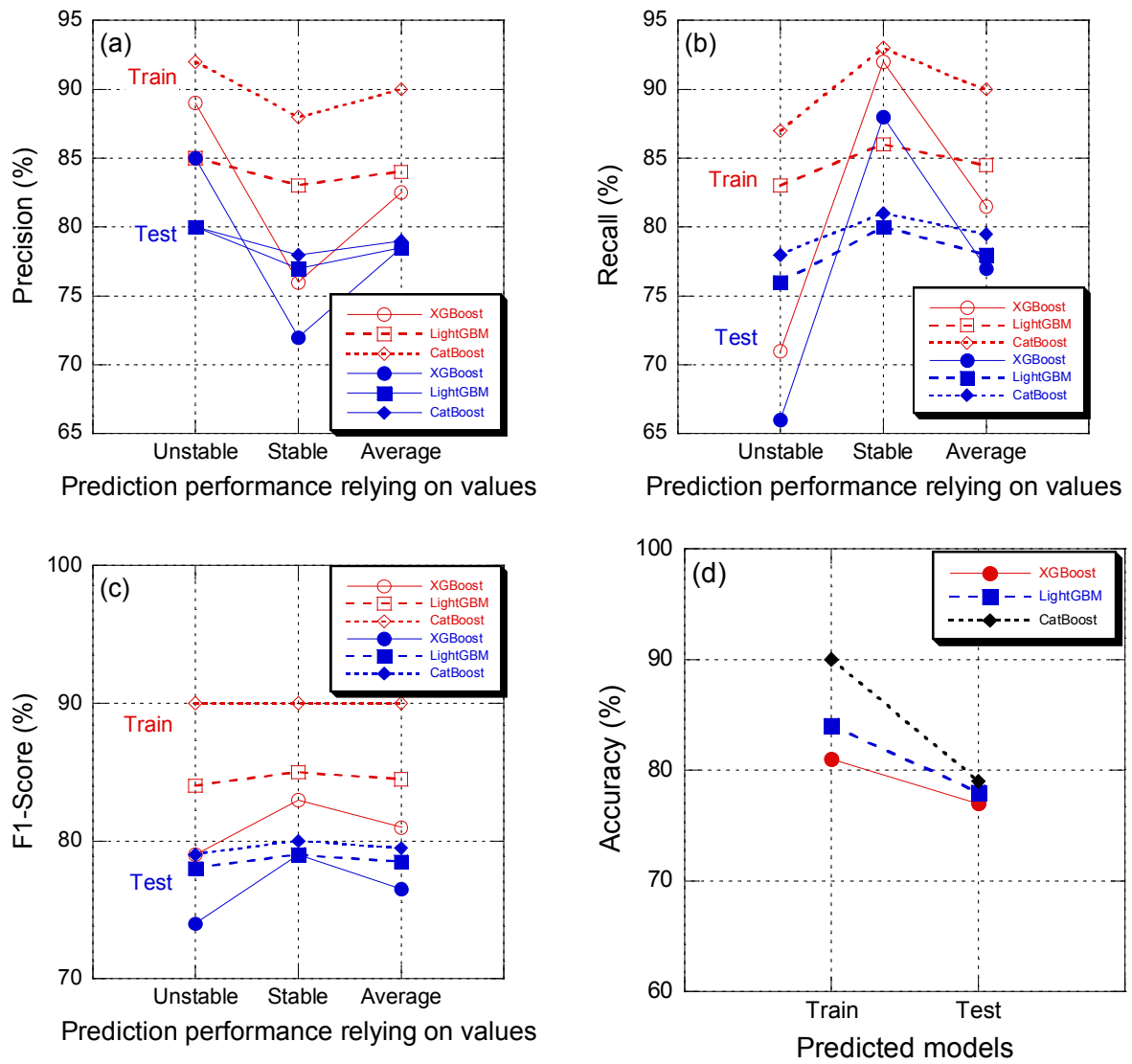


Figure 5-3. Result for prediction performance of slope stability model using Case-3 depending on machine learning methods (a) precision, (b) recall, (c) F1-Score (d) accuracy.

(4) 머신러닝 모듈에 따른 비탈면 안정성 예측 모델의 비교분석

XGBoost, LightGBM, CatBoost 등 각 모듈에 따른 비탈면 안정성 예측 모델을 분석한 결과, XGBoost를 이용한 비탈면 안정성 예측 모델에서 양호한 AUC, 예측 성능을 보였지만, 정밀도와 재현율에 불균형 현상이 크게 나타났다. LightGBM을 이용한 비탈면 안정성 예측 모델에서는 양호한 AUC, 학습 성능, 예측 성능을 나타냈다. CatBoost를 이용한 비탈면 안정성 예측 모델에서는 머신러닝 모델 중에서 가장 큰 AUC와 예측 성능을 보였지만, 범주형 데이터가 포함된 해석 단계에서 훈련 데이터에 대해 심한 과적합 현상이 나타났다.

변수 중요도는 XGBoost의 경우 범주형 데이터보다 수치형 데이터가 상대적으로 높은 값을 나타냈고, 수치형 데이터 중에서는 ‘소단 개소’와 ‘계곡부 개소’를 제외한 모든 수치형 데이터가 높게 나타났다. 범주형 데이터 중에서는 ‘암반형태’, ‘주변지형’, ‘지하수’의 변수 중요도가 높게 나타났다. LightGBM의 경우 XGBoost와 마찬가지로 범주형 데이터보다 수치형 데이터의 변수 중요도가 상대적으로 높은 값을 보였으며, 수치형 데이터 중에서도 ‘소단 개소’와 ‘계곡부 개소’를 제외한 모든 수치형 데이터가 높게 나타났다. 범주형 데이터 중에서는 ‘암반형태’, ‘주변지형’의 변수 중요도가 높게 나타났다. CatBoost의 경우 수치형 데이터보다 범주형 데이터의 변수 중요도가 상대적으로 높게 나타났다. 수치형 데이터 중에서는 ‘토층심도’, 범주형 데이터 중에서는 ‘불연속면 방향성’, ‘측면형상’, ‘주변지형’의 변수 중요도가 높게 나타났다. XGBoost와 LightGBM은 수치형 데이터의 변수 중요도가 높게 나타났고, CatBoost의 경우 범주형 데이터의 변수 중요도가 높게 나타났다.

XGBoost, LightGBM, CatBoost 모델의 모든 해석 단계에 대한 예측 성능이 선행연구들의 예측 성능 보다 높은 수치인 60% 후반의 준수한 예측 성능을 나타냈다. 하지만 XGBoost의 경우 모든 해석 단계에서 정밀도와 재현율의 불균형이 발생하였다. LightGBM의 경우 모든 해석 단계에서 양호한 학습 성능과 예측 성능을 나타냈다. CatBoost의 경우 모든 해석 단계에서 3개의 머신러닝 모델 중에서 가장 좋은 학습 성능과 예측 성능을 나타냈지만, 학습데이터에 과적합이 발생하였다. 따라서, XGBoost, LightGBM, CatBoost 모두 비탈면 안정성 예측 모델로서 사용 가능 할 것으로 사료되지만, 이들 중 LightGBM이 가장 안정적으로 비탈면 안정성을 예측할 수 있을 것으로 판단되고, CatBoost는 가장 정확하게 비탈면 안정성을 예측할 것으로 사료된다.

5.2 비탈면 조사자료의 시각화를 통한 비탈면 관리 방안

전국 비탈면 현장 조사자료를 이용해 ‘붕괴유형’과 ‘사면길이’를 일정 간격으로 분할하여 시각화하여 나타내면 Figure 5-4 및 Figure 5-5와 같다. Figure 5-4는 전국 비탈면 현장 조사자료를 이용하여 사면의 안정성을 ‘붕괴’와 ‘안정’으로 구분하여 나타낸 것이다. 이것을 전라남도지역으로 한정하여 나타내면 그림의 오른쪽과 같다. Figure 5-5는 전국 비탈면의 사면길이를 50 m 범위로 5구간으로 구분하여 표시한 것으로서, 이를 전라남도지역만을 상세하게 나타내면 오른쪽과 같다. 이들 그림에서 보는 바와 같이 본 연구 결과물의 하나로써 비탈면 대상지역을 목적에 따라 광역적 및 국지적으로 시각화할 수 있으므로, 비탈면 현장 조사자료를 시각화하여 이용한다면 전국에 존재하는 비탈면을 보다 효율적이고 효과적으로 유지 및 관리할 수 있을 것으로 사료 된다. 특히, 출력데이터를 각 ‘필요주공법’, ‘조치’, ‘붕괴 발생과 관련한 인자’ 등으로 설정함으로써, 보강 공법 또는 후속 조치에 대한 판단이 가능할 것으로 판단된다. 예측 모델이 비탈면의 상태를 불안정으로 예측한 경우 보강 공법을 제안함으로써 분류의 약점인 비탈면의 안전율을 고려하지 않은 점을 보완 가능할 것으로 사료 된다. 더 나아가 비탈면의 위치 정보를 이용하여 현장 조사자료와 현장을 실시간으로 확인할 수 있는 위성자료, CCTV의 영상자료와의 교차 검증을 통해 현장 조사자료의 신뢰도 향상은 물론, 비탈면에 대해 실시간 평가와 관리도 가능할 것으로 판단된다.

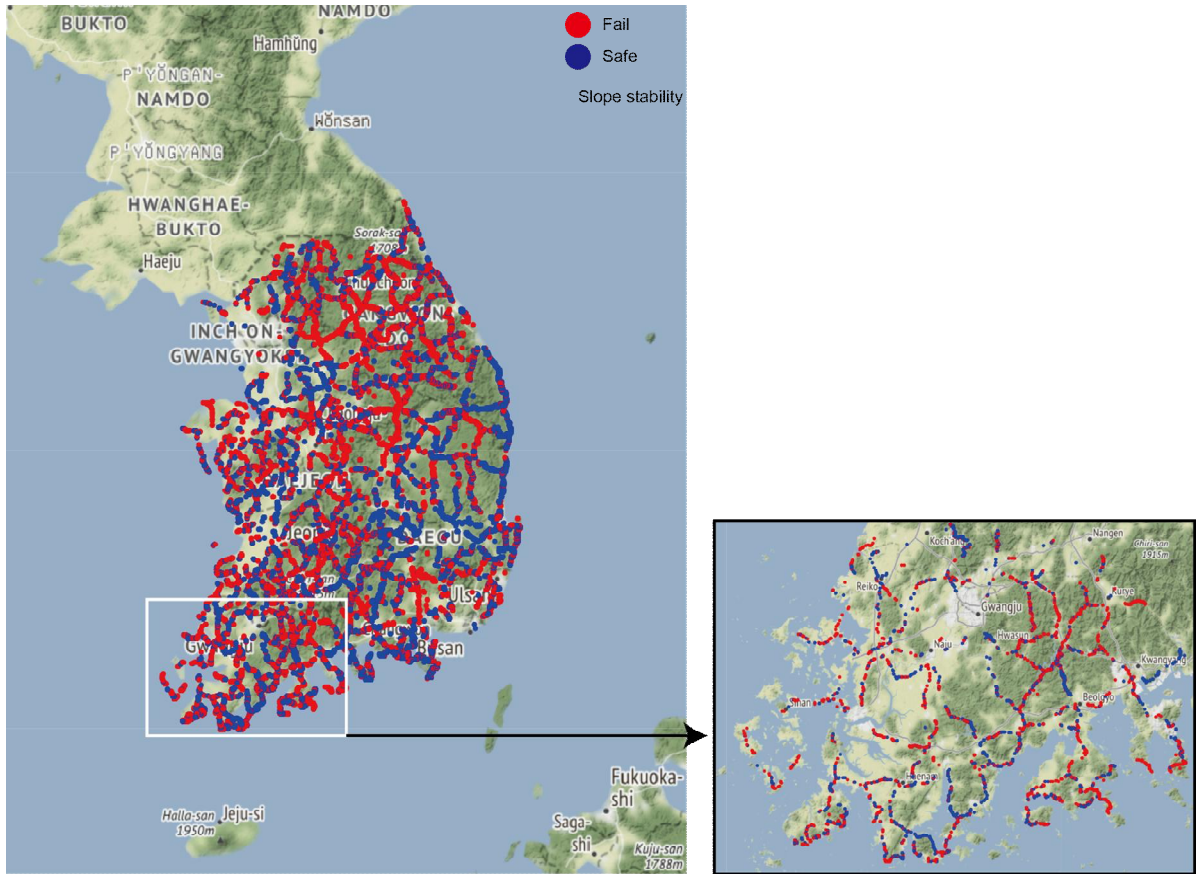


Figure 5-4. Visualization of slope stability through national slope investigation data.

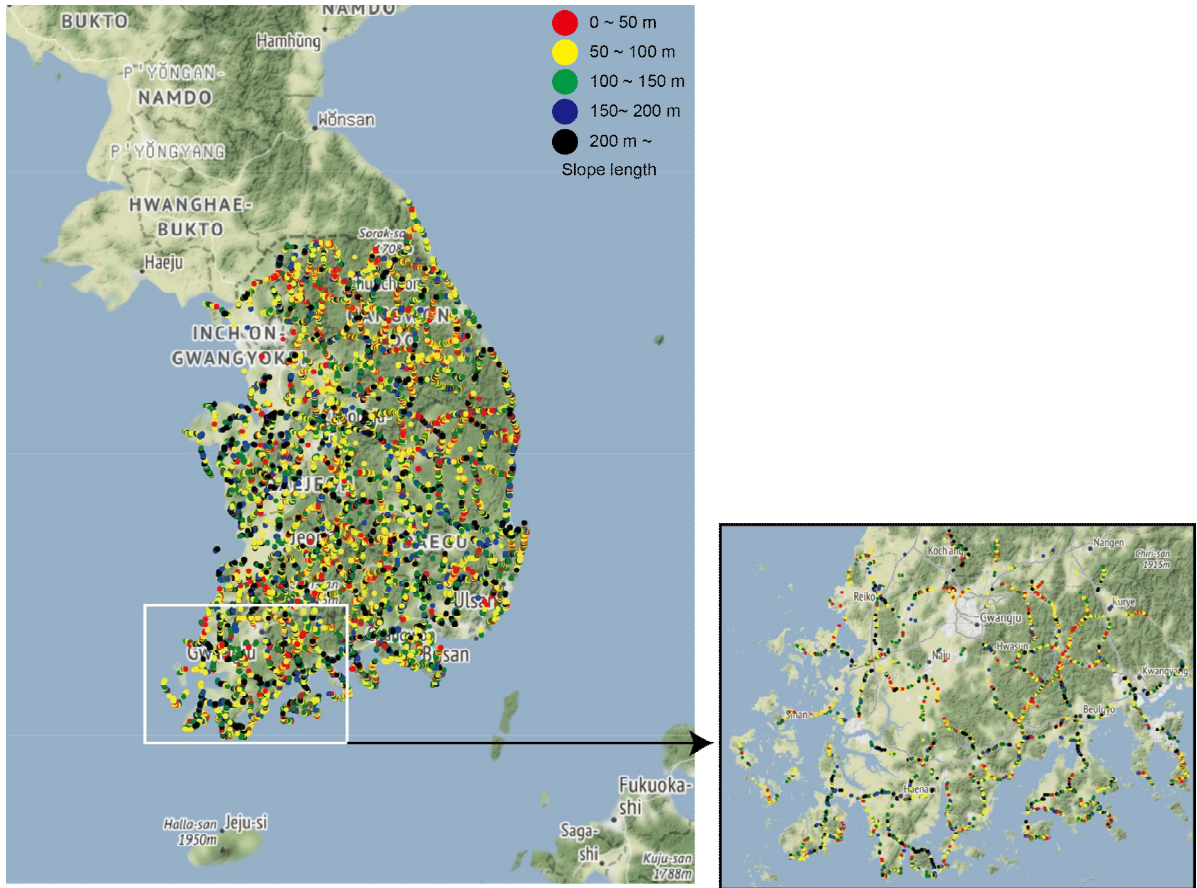


Figure 5-5. Visualization of slope length through national slope investigation data..

6. 결론

비탈면에 대한 안정성 예측 모델을 제안하기 위해 XGBoost, LightGBM, 그리고 CatBoost와 비탈면 조사자료를 활용하여 분류 모델을 생성하여 평가하였다. 이를 위해 비탈면 조사자료에 대해 결측치 및 이상치 제거, 변수 단순화 및 변환 등의 데이터 전처리 과정을 실시하였다. 또한 비탈면 안정성과 조사자료에 대한 통계적 검증과 상관 관계를 분석하여 예측 모델의 입력변수 31개와 출력변수 1개를 선정하였다. 마지막으로 비탈면 예측 모델을 크게 수치형 데이터 9개로 이루어진 Case-1, 범주형 데이터 22개로 이루어진 Case-2, 수치형 및 범주형 데이터 31개로 이루어진 Case-3 등 3가지 경우의 비탈면 안정성 예측 모델에 대한 비교 분석을 수행하였다. 모델 훈련에 사용한 훈련데이터와 시험데이터의 비율은 7:3으로 설정하였고, 초매개변수 최적화는 베이지안 최적화 방법을 사용하였다. 훈련데이터에 과적합하는 현상을 방지하기 위해 계층별 k-fold 교차 검증을 5 분할하여 실시하였다. 각 해석 단계에 대한 모델 결과는 학습 성능과 예측 성능, 그리고 변수 중요도를 산정하여 비교 분석하였다.

XGBoost를 활용하여 비탈면 안정성 예측 모델을 분석한 결과, AUC는 Case-1의 경우 0.668, Case-2의 경우 0.771, Case-3의 경우 0.770으로 나타났다. Case-1에 대한 예측 모델의 학습 성능은 훈련데이터와 시험데이터가 유사한 학습 성능을 나타냈으며, 훈련데이터와 시험데이터에서 F1-Score와 정확도가 각각 68%, 67%로 비슷한 예측 성능을 나타냈다. 정밀도와 재현율은 약 8~9%의 차이를 보였다. 변수 중요도는 ‘소단 개소’와 ‘계곡부 개소’를 제외한 ‘사면길이’, ‘사면경사’ 등의 변수들은 비슷한 중요도를 나타냈다. Case-2에 대한 예측 모델의 학습 성능은 훈련데이터에 약한 과적합하려는 현상을 보였다. 예측 성능 결과 훈련데이터와 시험데이터의 정확도와 F1-Score가 각각 83%, 77%로 약 5%의 차이를 보였으며, 정밀도와 재현율은 약 10%의 예측 성능 불균형을 나타냈다. 변수의 중요도는 ‘주변지형’, ‘암종’, ‘암반형태’, ‘비탈면 보강공법 2번의 공법 종류’, ‘풍화도’가 다른 변수들보다 상대적으로 높게 나타났다. Case-3에 대한 예측 모델의 학습 성능은 Case-2와 마찬가지로 훈련데이터에 약한 과적합하려는 현상을 보였다. 예측 성능 결과 훈련데이터와 시험데이터의 정확도와 F1-Score가 각각 81%, 77%로 약 5%의 차이를 보였으며, 정밀도와 재현율은 16~19%의 예측 성능 불균형을 나타냈다. 변수의 중요도

는 ‘사면길이’, ‘사면상부경사’, ‘토층심도’ 등 ‘소단 개소’와 ‘계곡부 개소’를 제외한 수치형 데이터가 높게 나타났다.

LightGBM을 활용한 비탈면 안정성 예측 모델을 분석한 결과, AUC는 Case-1의 경우 0.681, Case-2의 경우 0.766, Case-3의 경우 0.783으로 나타났다. Case-1에 대한 예측 모델의 학습 성능은 훈련데이터와 시험데이터가 유사한 학습 성능을 나타냈으며, 정밀도, 재현율, F1-Score, 정확도가 각각 약 73%, 68%로 비슷한 예측 성능을 나타냈다. 변수 중요도는 ‘소단 개소’와 ‘계곡부 개소’를 제외한 ‘사면길이’, ‘사면상부경사’, ‘사면경사’ 등의 변수들은 비슷한 중요도를 보였다. Case-2에 대한 예측 모델의 학습 성능은 Case-1과 마찬가지로 훈련데이터와 시험데이터가 유사한 학습 성능을 나타냈다. 예측 성능은 정밀도, 재현율, F1-Score, 정확도가 훈련데이터와 시험데이터에서 각각 약 79%, 77%로 유사한 성능을 나타냈다. 변수 중요도는 ‘암반형태’, ‘주변지형’이 다른 변수들보다 상대적으로 크게 나타났다. Case-3에 대한 예측 모델의 학습 성능은 양호한 성능을 나타냈지만, 다른 해석 단계 Case-1, Case-2 보다 상대적으로 훈련데이터에 과적합되는 양상을 보였다. 예측 성능 결과 훈련데이터와 시험데이터의 정밀도, 재현율, F1-Score, 정확도가 각각 84%, 78%로 5% 이하의 오차를 나타냈다. 변수 중요도는 ‘토층심도’, ‘사면상부경사’, ‘사면경사’, ‘사면길이’, ‘사면높이’, ‘이격거리’ 등의 변수들이 다른 변수들에 비해 상대적으로 중요도가 높게 나타났다.

CatBoost를 활용한 비탈면 안정성 예측 모델을 분석한 결과, AUC는 Case-1의 경우 0.672, Case-2의 경우 0.777, Case-3의 경우 0.794으로 나타났다. Case-1에 대한 예측 모델의 학습 성능은 훈련데이터와 시험데이터가 유사한 학습 성능을 나타냈으며, 정밀도, 재현율, F1-Score가 훈련데이터와 시험데이터에서 각각 약 70%, 67%로 비슷한 예측 성능을 나타냈다. 훈련데이터를 사용한 예측 모델과 시험데이터를 사용한 예측 모델의 정확도는 각각 70%, 66%로 4% 차이를 보였다. 변수 중요도는 ‘토층심도’, ‘사면상부경사’, ‘사면높이’ 순으로 중요도가 높게 나타났다. Case-2에 대한 예측 모델의 학습 성능은 훈련데이터에 과적합 현상을 보였다. 예측 성능 결과에서 훈련데이터와 시험데이터의 정확도는 각각 86%, 78%로 8%의 차이를 보였고, 정밀도와 재현율은 4~6%의 오차를 나타냈다. 변수 중요도는 ‘불연속면 방향성’, ‘측면형상’, ‘주변지형’ 등의 변수가 순서대로 높게 나타났다. Case-3에 대한 예측 모델의 학습 성능은 Case-2 보다 훈련데이터에 과적합하는 현상을 나타냈고, 훈련데이터와 시험데이터의 정확도는 각각 90%, 79%로 11%의 오차를

보였고, 정밀도, 재현율, F1-Score의 예측 성능 모두에서 약 5%의 차이를 나타냈다. 변수 중요도는 다른 머신러닝 모듈의 Case-3와 다르게 ‘토층심도’가 가장 크게 나타났고, 그다음 범주형 데이터인 ‘불연속면 방향성’, ‘측면형상’, ‘불연속면 종류’ 등의 변수들이 순서대로 높게 나타났다.

XGBoost은 전체적으로 양호한 AUC와 학습 성능을 나타냈지만, 약한 예측 불균형이 나타났다. 학습 성능의 경우 Case-1에서 훈련 및 시험데이터에서 좋은 결과를 나타냈지만, Case-2와 3에서 훈련데이터에 약한 과적합 현상을 보였다. 변수의 중요도는 사면의 형상을 나타내는 수치형 데이터들이 상대적으로 높게 나타났으며, 범주형 데이터 중에서는 ‘암반형태’, ‘주변지형’, ‘지하수’ 등이 높게 나타났다. LightGBM은 XGBoost와 다르게 3가지 해석 방법에서 예측 불균형을 보이지 않았으며, 학습 성능에서도 훈련데이터와 시험데이터에서 유사한 양상을 나타냈다. 변수의 중요도는 수치형 데이터가 상대적으로 높게 나타났고, 범주형 변수 중에서는 ‘암반형태’, ‘주변지형’이 높게 나타났다. CatBoost는 학습 성능에 있어서 Case-1을 제외하고 다른 모델들에 비해 훈련데이터에 과적합 현상이 크게 나타났다. 변수 중요도는 다른 모델들과 다르게 수치형 데이터보다 범주형 데이터가 상대적으로 높게 나타났다. 변수 중요도의 최대값은 ‘토층심도’지만 다른 상위 변수들은 ‘불연속면 방향성’, ‘측면형상’, ‘불연속면 종류’ 등의 범주형 변수로 나타났다.

본 연구의 결과를 종합해 볼 때, 모델의 예측 성능 평가를 기준으로 수치형 또는 범주형 데이터로 구성된 각각의 모델들은 단독으로 비탈면 안정성 예측 모델로 사용 가능할 것으로 판단된다. 또한 예측 성능을 기준으로 비탈면 안정성을 평가하는데 수치형 데이터보다 범주형 데이터가 더 중요한 역할을 할 것으로 판단된다. 하지만, 학습 성능에 있어서 범주형 데이터는 과적합 발생 가능성도 함께 나타날 수 있으므로 데이터 사용 시 이에 대한 충분한 고려도 있어야 할 것으로 사료된다. 결과적으로 본 연구에서 사용된 XGBoost, LightGBM, CatBoost 등은 비탈면 안정성을 예측하기 위한 모델로서 충분히 사용 가능할 것으로 판단된다. 이들 모델 중 LightGBM은 가장 안정적으로 비탈면 안정성을 예측할 수 있는 모델, CatBoost는 가장 정확하게 비탈면 안정성을 예측할 수 있는 모델로서 이용될 수 있을 것으로 사료된다.

참고문헌

Al Daoud, E., 2019, Comparison between XGBoost, LightGBM and CatBoost using home credit dataset, *International Journal of Computer and Information Engineering*, 13, 1, 6-10.

An, K. M., 2021, Developing a prediction model for firm innovation and performance using statistical matching and machine learning ensemble techniques, Dongguk University, Doctoral dissertation.

Bui, D. T., Ho, T. C., Pradhan, B., Pham, B. T., Nhu, V. H., Revhaung, I., 2016, GIS-based modeling of rainfall-induced landslides using data mining-based functional trees classifier with Adaboost, Bagging, and MultiBoost ensemble frameworks, *Environmental Earth Sciences*, 75, 1101.

Caine, N., 1980, The rainfall intensity: Duration control of shallow landslides and debris flow, *Mediterranean Storms: Proceedings of the EGS 2nd plinius Conference*, Siena, Italy, 463-487.

Cha, A. R., Kim, T. H., Gang, S. K., 2015, Application of geomorphological features for establishing the preliminary landslide hazard, *Journal of the Korean Society for Geospatial Information Science*, 23, 3, 23-29.

Chen, F., Yu, B., Li, B., 2017, A practical trial of landslide detection from single-temporal Landsat8 images using contour-based proposals and random forest: a case study of national Nepal, 15, 453-464.

Chen, T., Guestrin, C., 2016, XGBoost: A scalable tree boosting system, In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Newyork, USALACM, 785-794, DOI:10.1145/2939672.2939785

Cheon, M. J., Choi, H. J., Park, J. W., Choi, H. J., Lee, D. H., Lee, O., 2021, A study on the traffic flow prediction through Catboost algorithm, *Journal of the Korea Academia-Industrial cooperation Society*, 22, 3, 58-64.

Chung, C. F., Fabbri, A. G., 1999, Probabilistic prediction models for landslide hazard mapping, *Photogrammetric Engineering and Remote Sensing*, 65, 1389-1399.

Cho, T. H., 2017, *Deep learning for everyone*, second edition, gilbut, 1-368.

Dao, D. V., Jaafari, A., Bayat, M., Mafi-Gholami, D., Qi, C., Moayedi, H., Phong, T. V., Ly, H. B., Le, T. T., Trinh, P. T., Luu, C., Quoc, N. K., Thanh, B. N., Pham, B. T., 2020, A spatially explicit deep learning neural network model for the prediction of landslide susceptibility, *Catena*, 188, 104451.

Di, B., Zhang, H., Liu, Y., Li, J., Chen, N., Stamatopoulos, C. C., Luo, Y., Zhan, Y., 2019, Assessing susceptibility of debris flow in Southwest China using gradient boosting machine, *Scientific Reports* 9, 12532.

DSBA, [Korea University School of industrial management engineering DSBA laboratory], 2020a, November 12, 04-7: Ensemble Learning - XGBoost, Youtube, https://www.youtube.com/watch?v=VHky3d_qZ_E.

DSBA, [Korea University School of industrial management engineering DSBA laboratory], 2020b. November 17, 04-8: Ensemble Learning - LightGBM, Youtube, <https://www.youtube.com/watch?v=4C8SUZJPIMY>.

DSBA, [Korea University School of industrial management engineering DSBA laboratory], 2020c. November 20, 04-9: Ensemble Learning - CatBoost, Youtube, https://www.youtube.com/watch?v=2Yi_Jse_7JQ.

Fausset, L., 1994, *Fundamentals of neural networks: architectures, algorithms, and applications*, Prentice Hall, 1-461.

Frazier, P. I., 2018, A tutorial on bayesian optimization, arXiv preprint arXiv:1807.02811.

Freund, Y., Schapire, R. E., Naoki, A., 1999, A short introduction to boosting, *Journal-Japanese Society For Artificial Intelligence*, 14, 5, 771-780

Gao, W., Dai, S., Chen, X., 2019, Landslide prediction based on a combination intelligent method using the GM and ENN: two cases of landslides in the Three Gorges Reservoir,

Chian, Landslides, 17, 111-126.

Getachew, N., Meten, M., 2021, Weights of evidence modeling for landslide susceptibility mapping of Kabi-Gebro locality, Gundomeskel area, Central Ethiopia, *Geoenvironmental Disasters*, 8, 6.

Gomez, H., Kavzoglu, T., 2005, Assessment of shallow landslide susceptibility using artificial neural networks in Jabonosa River Basin, Venezuela, *Engineering Geology*, 78, 11-27.

Guzzetti, F., Reichenbach, P., Ardizzone, F., Cardinali, M., Galli, M., 2006, Estimating the quality of landslide susceptibility models, *Geomorphology*, 81, 166-184.

Guzzetti, F., Peruccacci, S., Rossi, M., Stark, C., 2007, Rainfall thresholds for the initiation of landslides in central and southern Europe, *Meteorology and Atmospheric Physics*, 98, 239-267.

Hong, M. H., Kim, J. H., Jung, G. J., Jeong, S. S., 2016, Rainfall Threshold (ID curve) for Prediction Considering Antecedent Rainfall, *Journal of the Korea Geotechnical Society*, 32, 4, 15- 27.

Hong, M., Jeong, S., Kim, J., 2019, A combined method for modeling the triggering and propagation of debris flows, *Landslides*, 17, 805-824.

Huang, F., Zhang, J., Zhou, C., Wang, Y., Huang, J., Zhu, L., 2019, A deep learning algorithm using a fully connected sparse autoencoder neural network for landslide susceptibility prediction, *Landslides*, 17, 217-229.

Kang, H. J., 2018, Optimization of multi-well placement using artificial intelligence, Master's thesis, Chosun University.

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., Liu, T. Y., 2017, LightGBM: A highly efficient gradient boosting decision tree, *Advances in neural information processing system*, 30.

Kim, E. M., 2020, Comparison of prediction models using machine learning of housing

tenure and decision-making on housing sales in the Korean Housing Market, Doctoral dissertation, Hansung University.

Kim, J. C., Lee, S. M., Jung, H. S., Lee, S., 2017, Landslide susceptibility mapping using random forest and boosted tree models in Pyeong-Chang, Korea, Geocarto International, 33, 9, 1000-1015.

Kim, J. H., Koo, H. B., Rhee, J. H., Yun, C. J., 2008, Study on Dangerous Cut-Slopes Distribution Using Inventory Data in Chungcheongdo, Korean Geotechnical Society, Proceedings of the Korean Geotechnical Society Conference, 858-862.

Kim, M. I., Lee, S. W., Kim, B. S., 2017, Slope stability assessment induced by variation in mountain topography and rainfall infiltration, The Journal of Engineering Geology, 27, 2, 125-132.

Kim, K. T., Jung, S. G., Park, K. H., Oh, J. H., 2005, Evaluation of landslide susceptibility using GIS and RS, Journal of the Korean Association of Geographic Information Studies, 8, 1, 75-87.

Kim, S. J., 2021, A study on malware classification method based on Light Gradient Boosting model, Master's thesis, Soonchunhyang University.

Kim, S. P., 2016, The first step in deep learning, hanbit media, 1-196

Kim, S. W., Chun, K. W., Kim, M., Catani, F., Choi, B., Seo, J. I., 2020, Effect of antecedent rainfall conditions and their variations on shallow landslide-triggering rainfall thresholds in South Korea, Landslides, 18, 569-582.

Ko, S. M., Lee, S. W., Yune, C. Y., Kim, G., 2014, Topographic analysis of landslide in Umyeonsan, Journal of the Korean Society of Surveying, Geodesy, Photogrammetry and Cartography, 32, 1, 55-62.

Korea Forest Service, 2021, comprehensive measures for national landslide prevention, 1-56.

Krkac, M., Gazibara, S. B., Arbanas, Z., Secanj, M., Arbanas, S. N., 2020, A comparative study of random forests and multiple linear regression in the prediction of landslide velocity, *Landslides*, 17, 2515-2531.

Kwak, J. H., Kim, M. I., Lee, S. J., 2018, Landslide susceptibility assessment considering the saturation depth ratio by rainfall change, *The Journal of Engineering Geology*, 28(4), 687-699.

Lee, H., Kim, G., 2012, Landslide risk assessment in Inje using logistic regression model, *Journal of the Korean Society of Survey, Geodesy, Photogrammetry, and Cartography*, 30, 3, 313-321.

Lee, S., Jeong, G., Park, S. J., 2015, Evaluating geomorphological classification systems to predict the occurrence of landslides in mountainous region, *Journal of the Korean Geographical Society*, 50, 5, 485-503.

Lee, S. H., Kim, S. H., Woo, Y., Moon, J. P., Yang, I., 2021, The study for improvement of data-quality of cut-slope management system using machine learning, *The Journal of Engineering Geology*, 31, 1, 31-42.

Lee, S. R., Chawe, U. C., Chang, B. S., 2002, Development and application of landslide analysis technique using geological structure, *The Journal of GIS association of Korea*, 10, 2, 247-261.

Lee, Y. K., 2021, Optimization of machine learning algorithms for land use and land cover classification using remotely-sensed data, Kangwon National University, Master's thesis.

Liu, Y., Wu, L., 2018, High performance geological disaster recognition using deep learning, *Precedia Computer Science*, 129, 529-536.

Melchiorre, C., Matteucci, M., Azzoni, A., Zanchi, A., 2008, Artificial neural networks and cluster analysis in landslide susceptibility zonation, *Geomorphology* 94, 379-400.

MOLIT (Ministry of Land, Infrastructure and Transport), 2018, Road slope maintenance practical manual (11-1613000-002385-01)

Nam, K., Kim, M. I., Kwon, O., Wang, F., Jeong, G. C., 2020, Prediction of landslides and determination of its variable importance using AutoML, *The Journal of Engineering Geology*, 30, 3, 315-325.

Natekin, A., Knoll, A., 2013, Gradient boosting machines, a tutorial, *Frontier in neurorobotics*, 7, 21.

Nhu, V.H., Zandi, D., Shahabi, H., Chapi, K., Shirzadi, A., Al-Ansari, N., Singh, S. K., Dou, J., Nguyen, H., 2020, Comparison of support vector machine, bayesian logistic regression, and alternating decision tree algorithms for shallow landslide susceptibility mapping along a Mountainous Road in the West of Iran, *Applied Sciences*, 10, 15, 5047.

Nhu, V. H., Mohammadi, A., Shahabi, H., Ahmad, B. B., Al-Ansari, N., Shirzadi, A., Clague, J. J., Jaafari, A., Chen, W., Nguyen, H., 2020, Landslide susceptibility mapping using machine learning algorithms and remote sensing data in a tropical environment, *International Journal of Environmental Research and Public Health*, 17, 14, 4933.

Oh, J., Park, H. J., 2014, Analysis of landslide triggering rainfall threshold for prediction of landslide occurrence, *J. Korean Soc. Hazard Mitig.*, 14, 2, 115-129.

Oh, H. T., 2020, A study on the machine learning model for the production trend forecast of oil and gas wells, Master's thesis, Chosun University.

Oilvares, G., Escalona, C., and Gimenez, E., 2012, Production monitoring using artificial intelligence, APLT Asset, SPE 149594, Presented at SPE Intelligent Energy International held in Utrecht, The Netherlands, 27-29, March 2012.

Pham, B. T., Pradhan, B., Bui, D. T., Prakash, I., Dholakia, M. B., 2016, A comparative study of different machine learning methods for landslide susceptibility assessment: A case study of Uttarakhand area(India), *Environmental modelling & Software*, 84, 240-250.

Pradhan, S. M. S., Lee, S. R., Kim, Y. T., 2018, A shallow slide prediction model combining rainfall threshold warnings and shallow slide susceptibility in Busan, Korea, *Landslides*, 16, 647-659.

Pradhan, B., Lee, S., Buchroithner, M. F., 2010, A GIS-based back-propagation neural network model and its cross-application and validation for landslide susceptibility analyses, *Computers, Environment and Urban Systems*, 34, 3, 216-235.

Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., Gulin, A., 2017, CatBoost: unbiased boosting with categorical features, 32nd *Conferences on Neural Information Processing Systems*, Montreal, Canada, 31.

Saito, H., Nakayama, D., Matsuyama, H., 2009, Comparison of landslide susceptibility based on a decision-tree model and actual landslide occurrence: The Akaishi Mountains, Japan, *Geomorphology*, 109, 108-121.

Sahin, E. K., 2020, Comparative analysis of gradient boosting algorithms for landslide susceptibility mapping, *Geocarto International*, DOI:10.1080/10106049.2020.1831623

Schapire, R. E., 1999, A brief introduction to boosting, In *Ijcai*, 99, 1401, 1406.

Seo, W. G., Choi, J., Chae, B. G., Song, Y. S., 2017, Characteristics of landslide occurrence and change in the matric suction and volumetric water content due to rainfall infiltration, *The Journal of Engineering Geology*, 27, 4, 475-487.

Skilodimou, H. D., Bathrellos, G. D., Koskeridou, E., Soukis, K., Rozos, D., 2018, Physical and anthropogenic factors related to landslide activity in the Northern Peloponnese, Greece, *Land*,

Snoek, J., Hugo, L., Ryan, P. A., 2012, Practical bayesian optimization of machine learning algorithms, *Advances in neural information processing systems*, 2.

Su, Q., Zhang, J., Zhao, S., Wang, L., Liu, J., Guo, J., 2017, Comparative assessment of three nonlinear approaches for landslide susceptibility mapping in a Coal Mine Area, *International Journal of Geo-Information*, 6, 7, 228.

Suzen, M. L., Doyuran, V., 2004, Data driven bivariate landslide susceptibility assessment using geographical information system: a method and application to Asarsuyu catchment, Turkey, *Engineering Geology*, 71, 303-321.

USGS, The Landslide handbook, A guide to understanding landslides, 1-126.

Wang, L. H., Guo, M., Sawada, K., Lin, J., Zhang, J., 2016, A comparative study of landslide susceptibility maps using logistic regression, frequency ratio, decision tree, weights of evidence and artificial neural network, Geosciences Journal, 20, 1, 117-136.

Wang, H., Zhang, L., Yin, K., Luo, H., Li, J., 2021, Landslide identification using machine learning, Geoscience Frontiers, 12, 1, 351-364.

Wieczorek, G. F., 1985, Preparing a detailed landslide-inventory map for hazard evaluation and reduction, Environmental and Engineering Geoscience, 21, 3, 337-342.

Wikipedia, 2022a, Bayesian optimization, Retrieved from https://en.wikipedia.org/wiki/Bayesian_optimization.

Wikipedia, 2022b, Ensemble learning, Retrieved from https://en.wikipedia.org/wiki/Ensemble_learning.

Wikipedia, 2022c, Gradient boosting, Retrieved from https://en.wikipedia.org/wiki/Gradient_boosting.

Wikipedia, 2022d, Landslide, Retrieved from <https://en.wikipedia.org/wiki/Landslide>.

Woo, Y., Kim, S. H., Yang, I., Lee, S. H., 2020a, The study for utilizing data of cut-slope management system by using logistic regression, The Journal of Engineering Geology, 30, 4, 649-661.

Woo, Y., Kim, S. H., Kim, J. U., Park, G., 2020b, Research on the production of risk maps on cut slope using weather information and Adaboost model, The Journal of Engineering Geology, 30, 4, 663-671.

Xiao, T., Segoni, S., Chen, L., Yin, K., Casagli, N., 2020, A step beyond landslide susceptibility maps: a simple method to investigate and explain the different outcomes obtained by different approaches, Landslides, 17, 627-640.

Xu, Z., Kwak, H., Lee, W. K., Lee, W., K., Park, T., Kwon, T. H., Park, S., 2011, Assessment of landslide on climate change using GIS, *Journal of Climate Change Research* 2, 1, 43-54.

Yang, B., Yin, K., Lacass, S., Liu, Z., 2019, Time series analysis and long short-term memory neural network to predict landslide displacement, *Landslides*, 16, 677-694.

Yang, Z., Wang, L., Qiao, J., Uchimura, T., Wang, L., 2020, Application and verification of a multivariate real-time early warning method for rainfall-induced landslides: Implication for evolution of landslide-generated debris flows, *Landslides*, 17, 2049-2419.

Yeon, Y. K., 2011, Evaluation and analysis of Gwangwon-do landslide susceptibility using logistic regression, *Journal of the Korean Association of Geographic Information Studies*, 14, 4, 116-127.

Yoon, D., Kim, S. M., Kim, J., Park, G., Park, H., Byeon, J., Suh, J., Lee, C. W., Jang, I., Cho, S. J., Choi, Y., 2018, Introduction to resource development and machine learning, *CIR*, 1-460.

Yoon, H. R., 2020a, A empirical study on the financial stability prediction model of South Korea's public enterprises with machine learning techniques, Hansung University, Doctoral dissertation.

Yoon, G. N., Shin, H., Kim, Y. T., 2015, Effect of antecedent rainfall on infiltration characteristics in unsaturated soil, *Journal of the Korean Geotechnical Society*, 31, 8, 5-15.

Yoon, Y. G., 2020b, Feature extraction and analysis of electrocardiogram using LightGBM, Master's thesis, Korea University.

Zhou, C., Yin, K., Cao, Y., Intrieri, E., Ahmed, B., Catani, F., 2018, Displacement prediction of step-like landslide by applying a novel kernel extreme learning machine method, *Landslides*, 15, 11, 2211-2225.

감사의 글

학부 3학년 때부터 지금까지 약 10년간의 연구실 생활이 마무리되고 있습니다. 그동안 저에게 많은 도움을 주신 분들께 짧은 글로나마 감사의 말씀을 드립니다.

먼저 지금까지 부족한 저를 이끌어주신 인생의 멘토 강성승 교수님께 진심으로 감사드립니다. 교수님께 많은 것을 느끼고 배웠습니다. 아직도 많이 부족하다고 생각합니다. 안주하지 않고, 더 정진하여 자랑스러운 제자가 되겠습니다. 그리고 바쁘신 와중에도 부족한 저의 논문이 더 좋은 논문이 될 수 있도록 심사해주신 장일식 교수님, 서용석 교수님, 김대현 교수님, 나태유 박사님께도 진심으로 감사드립니다. 또한 학부 시절부터 지금까지 관심과 애정을 갖고 열정적으로 지도해주신 고진석 교수님, 최태진 교수님, 박철현 교수님, 박준규 교수님께도 감사의 말씀을 전해드리고 싶습니다.

에너지자원공학과를 다니며 맺었던 많은 인연 중에서 세심하게 많은 것을 알려줬던 우리 자원개발·암석역학 실험실의 영호형과 찬우형, 항상 응원해주셔서 감사합니다. 그리고 같이 열심히 공부했던 내 동기 마루와 주호, 동생들이지만 배울 점 많고 항상 최선을 다해 도와줬던 지현이, 건주, 정원이, 열심히 일하고 있는 채훈이와 남현이, 박사과정 동안 힘든 일 있을 때 많이 도와주셨던 성환이형과 홍찬이형에게도 감사의 인사 전합니다. 또한, 다른 연구실이지만 많은 도움을 준 석유공학 실험실의 영경이형, 현승이, 주환이, 유미, 득선이, 응용광물 실험실의 현수, 타국에서 공부하러 온 분지해석 실험실의 반로가, 바이오에너지공학 실험실의 희원이와 종화에게도 고맙다는 말 전하고 싶습니다. 이 외에도 현재 각자의 자리에서 열심히 일하고 있는 에너지자원공학과 선·후배님들 모두에게 감사의 인사 전합니다.

주변 지인 중에서 꿈을 위해 향해 공부하는 저를 응원하는 중학교 2학년 때부터 지금까지 좋은 일과 힘든 일 모두 함께 해줬던 정경이, 무리한 요구에도 흔쾌히 승낙해주는 영훈이, 내 디스코드 아이디 사용하는 은국이, 이제 막 아기 아빠가 된 병욱이, 관상 잘 보는 제욱이, 사이가 좋아 부러운 형제 상오형과 상암이, 듄직하게 말 잘 들어주는 진우, 이룬이 자랑하는 은안이, 목소리 큰 울로족 창호, 전화로 응원 많이 해주는 형우와 찬송이에게도 감사의 인사 전합니다.

마지막으로 지금까지 키워주시고 아낌없이 사랑해주시고, 제가 어떤 일이든 할 수 있게 만들어주는 우리 아빠, 엄마, 그리고 여러 가지 방법으로 지원 및 응원해

주는 누나에게 고맙고 사랑한다는 말을 전하고 싶습니다.

여러분들과 함께한 시간이 있기에 지금의 제가 있습니다. 모든 분들이 항상 건강하고 행복하길 기원합니다. 감사합니다.