



Wes Moore
Governor
Aruna Miller
Lieutenant Governor
Paul J. Wiedefeld
Secretary
William Pines, P.E.
Administrator

**MARYLAND DEPARTMENT OF TRANSPORTATION
STATE HIGHWAY ADMINISTRATION**

RESEARCH REPORT

**EFFECTIVELY IMPLEMENTING MACHINE LEARNING WITH
OFFICE OF MATERIALS TECHNOLOGY**

**Yunfeng Zhang, Professor, Principal Investigator
Jianshu Xu, Graduate Research Assistant
Nathan Moore, P.E., Technical Lead**

UNIVERSITY OF MARYLAND

FINAL REPORT

October 2023

This material is based upon work supported by the Federal Highway Administration under the State Planning and Research program. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the author(s) and do not necessarily reflect the views of the Federal Highway Administration or the Maryland Department of Transportation. This report does not constitute a standard, specification, or regulation.

TECHNICAL REPORT DOCUMENTATION PAGE

1. Report No. MD-21-SHA/UM/6-06	2. Government Accession No.	3. Recipient's Catalog No.	
4. Title and Subtitle Effectively Implementing Machine Learning with Office of Materials Technology		5. Report Date October 2023	
		6. Performing Organization Code Enter any/all unique numbers assigned to the performing organization, if applicable.	
7. Author(s) Dr. Yunfeng Zhang, Professor, University of Maryland (UMD) Jianshu Xu, Graduate Research Assistant, UMD Nathan Moore, P.E., Technical Lead, MDOT SHA		8. Performing Organization Report No. Enter any/all unique alphanumeric report numbers assigned by the performing organization, if applicable.	
9. Performing Organization Name and Address University of Maryland Department of Civil and Environmental Engineering College Park, Maryland 20742		10. Work Unit No.	
		11. Contract or Grant No. SHA/UM/6-06	
12. Sponsoring Agency Name and Address Maryland Department of Transportation (SPR) State Highway Administration Office of Policy & Research 707 North Calvert Street Baltimore MD 21202		13. Type of Report and Period Covered SPR-B Final Report (April 2021-May 2023)	
		14. Sponsoring Agency Code (7120) STMD - MDOT/SHA	
15. Supplementary Notes			
16. Abstract Machine learning models provide significant value for planning and asset management purposes by maximizing the usefulness of past data collection for decision-making and policy development. Newer and more powerful machine learning models not available in previous applications and research in highway transportation fields are investigated to address the challenging needs identified by OMT in design and construction, materials test data, management, operating optimization and field inspection. This research project is thus aimed to optimize and apply state-of-the-practice machine learning algorithms and application tools to address the emerging needs in the OMT needs for better operation and management of MD SHA assets. Specifically, machine learning models for landslide risk assessment, slope/embankment detection for geotechnical asset inventory, scarp line detection, concrete compressive strength test data modeling and prediction, pavement marking retroreflectivity data modeling and deterioration condition prediction, have been investigated in this project. There are many areas in the MDOT SHA Office of Materials Technology (OMT) that would greatly benefit from these valuable machine learning tools. Once the capability for model development is institutionalized, re-training by knowledgeable staff can enhance the models when datasets grow. Planning-level resources such as maps and estimated ranges of material properties, assist with preliminary engineering work. Phase II work is utilizing machine learning to support concrete strength predictions and geotechnical asset management efforts, including landslide risk, slope asset inventory development, and slope distress identification. As complex, structured data sets grow, accuracy will improve, along with increased reliance on machine learning analyses.			
17. Key Words Embankment, landslide risk, scarps, machine learning, neural networks, object detection, instance segmentation, pavement marking retroreflectivity, concrete test data		18. Distribution Statement This document is available from the Research Division upon request.	
19. Security Classif. (of this report) None	20. Security Classif. (of this page) None	21. No. of Pages 100	22. Price

TABLE OF CONTENTS

LIST OF FIGURES	iii
LIST OF TABLES	vi
CHAPTER 1: INTRODUCTION	
RESEARCH PROBLEMS & BACKGROUND	1
RESEARCH OBJECTIVES	2
RESEARCH APPROACH	3
CHAPTER 2: LIDAR DEM DATA DRIVEN MACHINE LEARNING MODELING OF HIGHWAY LANDSLIDE RISK	7
CHAPTER 3: INSTANCE SEGMENTATION MACHINE LEARNING MODEL FOR EMBANKMENT IDENTIFICATION	15
CHAPTER 4: INSTANCE SEGMENTATION MACHINE LEARNING MODEL FOR SLOPE SCARP IDENTIFICATION	33
CHAPTER 5: TABULAR NEURAL NETWORK MODEL DEVELOPMENT FOR CONCRETE STRENGTH TEST DATASETS	43
CHAPTER 6: MACHINE LEARNING MODEL DEVELOPMENT FOR PAVEMENT MARKING RETROREFLECTIVITY DATA	62
CHAPTER 7: TABULAR NEURAL NETWORK MODEL DEVELOPMENT FOR GEOTECHNICAL DRILLING DATA	72
CHAPTER 8: SUMMARY & CONCLUSIONS	94
REFERENCES	99

LIST OF FIGURES

Figure 1. Framework of raster-image based landslide risk machine learning model.....8

Figure 2. Lithology map of Maryland (1= low relief carbonates, 2 = low relief mudrock, 3= major ridge formers, 4= Moderate or variable quartzose ridge formers, 5= Moderate relief clastic rocks, 6= others, 7= Shaley units with interbedded sandstone) 10

Figure 3. Map of cost distance to streams of Anne Arundel County 11

Figure 4. landslide susceptibility map of western four counties (GA, AL, WA) in Maryland 12

Figure 5. Machine learning generated landslide risk map of Prince George County 13

Figure 6. Machine learning generated landslide risk map of Anne Arundel County..... 14

Figure 7. Flowchart of embankment and slope polygon detection using machine learning method 15

Figure 8. Slope composite band raster image of Washington County 17

Figure 9. Sample training chip with a 256x256 cell size for embankment detection model 19

Figure 10. Embankment polygon mapping with training chip rotation augmentation..... 19

Figure 11. Embankment polygons detected using the Instance Segmentation method at a location in Washington County along Highway I-70 20

Figure 12. Sample training loss curve for the embankment detection machine learning model..... 21

Figure 13. Comparison of machine learning generated embankment polygons with inventory data 21

Figure 14. Histogram of the filtered polygons from first-cycle machine learning detected embankment and slope polygons 22

Figure 15. Comparison of embankment polygon (a) before post-processing; (b) after post-processing 23

Figure 16. Joining the 1-ft transect line attributes with 1000-ft transect lines to assign each transect line with route mile point..... 24

Figure 17. Comparison of machine learning model detection embankment/slope polygon samples: (a) w/ 5000 chips randomly selected from three western counties (GA, AL and WA) rotation augmentation; (b) w/ all rotation augmentation training chips for three western counties; (c) without rotation augmentation chips..... 28

Figure 18. Slicing machine learning model detected embankment polygon with bridge linear feature: (a) Polygon before cutting by bridge buffer polygon; (b) Polygons after cutting 28

Figure 19. Embankment Detection by second round ML model: (a) detected polygons in 2nd round ML model; (b) training dataset from 1st round ML prediction 29

Figure 20. Sample embankment polygons detected from 2nd round machine learning model 30

Figure 21. Machine learning model detected embankment polygon post-processing: Attaching 1/1000th mile marker milepoints attribute 30

Figure 22. Training loss curve from training embankment detection machine learning model using the 1-m resolution Washington County LiDAR DEM data 31

Figure 23. Sample machine learning detected embankment and slope polygons using 1-m resolution LiDAR DEM data at Washington County..... 31

Figure 24. Flowchart of machine learning-enabled automated scarp polylines generation 33

Figure 25. Scarp line polygons with three different buffer width values (2m, 3m, 4m) for machine learning enabled scarp line detection 34

Figure 26. Illustration of the Slope tool in arc gis pro 35

Figure 27. Zonal statistics (maximum slope) histogram of scarp polygon’s slope values: (a) Results from arc gis Slope tool; (b) Results from arc gis Surface Parameter slope tool 36

Figure 28. Flowchart of instance segmentation machine learning model based scarp line detection using unscaled 8-bits unsigned 1-m surface parameter slope raster tiff data..... 36

Figure 29. Training loss curve for 3m-wide scarp line polygon detection machine learning model 37

Figure 30. Sample detected scarp line by the instance segmentation machine learning model (yellow polygon = detected scarp polygon, scarp line is the center line of the yellow polygon)..... 38

Figure 31. Training loss curve of scarp line detection machine learning model for the middle region counties..... 38

Figure 32. Machine learning detected scarp line samples in 4 western counties of Maryland 39

Figure 33. Maps of training chips from 11 middle region counties generated from 16-bits unsigned surface parameter slope raster tiff images 40

Figure 34. Samples of machine learning detected scarp lines in SM county..... 42

Figure 35. Samples of machine learning detected scarp lines (red lines = inventory scarp line data; blue or green line = machine learning detected scarp lines).....	42
Figure 36 Histogram of Mix 3 Cyl1 original data	48
Figure 37 Histogram of Mix 3 Cyl1 clean data	49
Figure 38 Histogram of Mix 3 Cyl2 Original data	49
Figure 39 Histogram of Mix 3 Cyl2 clean data	50
Figure 40 Histogram of Mix 6 Cyl1 original data	50
Figure 41 Histogram of Mix 6 Cyl1 Clean data	51
Figure 42 Architecture of feedforward neural network model for concrete strength data.....	51
Figure 43 Training loss curve, mix 6, model 1	51
Figure 44 Training loss curve, mix 6, model 2	52
Figure 45 x-y scatter plot mix 3, model 1	52
Figure 46 x-y scatter plot mix 3, model 2.....	52
Figure 47 x-y scatter plot mix 6, model 1	53
Figure 48 x-y scatter plot mix 6, model 2.....	53
Figure 49 Original distribution of Mix 6 Clean strength Mean CYL1 CYL2	54
Figure 50 Resampled distribution of Mix 6 strength Mean CYL1 CYL2	54
Figure 51 Original distribution of Mix 3 Clean strength Mean CYL1 CYL2	55
Figure 52 Resampled Mix 3 strength Mean CYL1 CYL2.....	55
Figure 53 Mix 3 model training loss curve.....	56
Figure 54 Mix 6 model training loss curve.....	56
Figure 55 x-y scatter plot for mix 6 mean cyl1 cyl2.....	57
Figure 56 x-y scatter plot for mix 3 mean cyl1 cyl2.....	57
Figure 57 SHAP plot for resampled dataset, mix 3	58
Figure 58. SHAP plot for original dataset, mix 6	58
Figure 59 SHAP plot for resampled dataset, mix 6	58
Figure 60 Training Loss Curve for mix 3 binary classification model.....	58
Figure 61 Training Loss Curve for mix 6 binary classification model.....	60
Figure 62 Accuracy report for mix 6 binary classification model	60
Figure 63 Mix 3 concrete real test strength vs. ML prediction confidence score scatter plot.....	61
Figure 64 Mix 6 concrete real test strength vs. ML prediction confidence score scatter plot.....	61
Figure 65. Geospatial visualization of pavement marking retroreflectivity dataset in arc gis pro.....	64
Figure 66. Visualization of 1/100th mile markers with application date (spatial distribution of 1/100 mile markers with application date and material features from condition data work sheet)	64
Figure 67. Maryland route network function class map	65
Figure 68. Maryland route network AADT (annual average daily traffic) map	65
Figure 69. Dataframe for pavement marking retroreflectivity machine learning model training	66
Figure 70. Geospatial distribution of pavement marking retroreflectivity data	67
Figure 71. MD-140 pavement marking retroreflectivity field test data matched to 1/100 mile makers	68
Figure 72. Training loss curve of white 40-mil pavement marking retroreflectivity machine learning model	68
Figure 73. Updated pavement marking retroreflectivity machine learning model predictions vs. true value (from field test) for pavement marking of White striping color	69
Figure 74. Training loss curve of machine learning model using Yellow 40-mil pavement marking retroreflectivity data	69
Figure 75. Updated pavement marking retroreflectivity machine learning model predictions vs. true value (from field test) for pavement marking of Yellow striping color	70
Figure 76. Feature importance ranking for white 40-mil pavement marking retroreflectivity machine learning model	70
Figure 77. MD grain size dataset distribution.....	77
Figure 78. Spatial distribution of Grain Size data (x-y plane).....	77
Figure 79. 3D spatial distribution of MD Grain Size Dataset.....	78
Figure 80. Concatenated grain size dataset distribution	78
Figure 81. SPT number buckets distribution	79
Figure 82. cleansed SPT number buckets distribution.....	79
Figure 83. Spatial distribution of SPTN bucket data (x-y plane).....	80
Figure 84. 3D spatial distribution of MD SPTN bucket Dataset	80

Figure 85. Histogram of MD ground water depth distribution	81
Figure 86. Histogram of USGS nationwide ground water depth distribution.....	81
Figure 87. Spatial distribution of USGS nationwide ground water depth distribution	82
Figure 88. Refusal depth bucket Distribution	82
Figure 89. Training loss curve for MD grain size model.....	83
Figure 90 Accuracy report for MD grain size model.....	83
Figure 91. Grain size predicted category - real category scatter plot.....	84
Figure 92. Training loss curve of consolidated grain size data model.....	84
Figure 93. Accuracy report of consolidated grain size data model.....	85
Figure 94. Training loss curve of MD groundwater depth dataset	85
Figure 95. x-y scatter plot of MD groundwater depth model	86
Figure 96. Training loss curve of USGS nationwide groundwater depth model	86
Figure 97. x-y scatter plot of USGS nationwide ground water depth model.....	87
Figure 98. Training loss curve of MD SPTN model.....	87
Figure 99. Accuracy report of MD SPTN model.....	88
Figure 100. Training loss curve of consolidated SPTN model	88
Figure 101. Accuracy report of consolidated SPTN model.....	89
Figure 102. Histogram of the resampled MD grain size dataset1	89
Figure 103. Histogram of the resampled MD grain size dataset1	90
Figure 104. Accuracy report of resampled grain size dataset model	90
Figure 105. Histogram of the resampled SPTN dataset.....	91
Figure 106. Accuracy report of resampled SPTN model.....	91
Figure 107. x-y scatter plot of resampled SPTN model.....	92
Figure 108. SHAP summary plot for MD grain size model	92
Figure 109. SHAP summary plot for consolidated grain size model.....	92
Figure 110. SHAP summary plot for SPTN model	93
Figure 111. SHAP summary plot for ground water depth model	93

LIST OF TABLES

Table 1. Computing information for Anne Arundel County landslide risk map generation	12
Table 2. Number of filtered polygons from first-cycle machine learning prediction appended to MDOT SHA inventory dataset to make the slope-type dataset more balanced for training	22
Table 3. Comparison of instance segmentation machine learning model detection results (confidence score ≥ 0.4)	26
Table 4. Quantity of embankment polygons after post-processing and filtering (confidence score ≥ 0.4)	27
Table 5. Quantity of machine learning detected scarp lines in 4 middle region counties (AA, BA, PG, CH) (confidence score ≥ 0.4)	40
Table 6. Quantity of machine learning detected scarp lines for nine counties (MO, HO, HA, CL, CA, CO, CH, FR, SM counties) (confidence score ≥ 0.3)	41
Table 7. Feature value range of concrete strength data for machine learning model training	44
Table 8. Reasonable range of pavement marking age (duration) and retro-reflectivity values	66
Table 9. Duration of pavement marking at retroreflectivity inspection time derived from the three installation-date data sources	67

CHAPTER 1: INTRODUCTION

RESEARCH PROBLEMS & BACKGROUND

Every year the Maryland State Highway Administration (SHA) invests millions of dollars into testing geomaterials to optimize engineering designs. There is a significant opportunity for cost savings by leveraging historic material testing data with predicative machine learning models to provide estimated values as well as gaining a better understanding of the historical data in the SHA inventory. In Phase 1 of this research project, Office of Materials Technology (OMT) has implemented machine learning in the following areas: predicting tabular data (drilling, pavement, test data), object recognition (guide rail end treatments, blurry or under-exposed photos, etc.) and reinforcement learning for drilling project schedule data sets (schedule estimation). By leveraging on this incredible progress, this Phase 2 project identified more datasets where machine learning would be beneficial such as LiDAR DEM data and pavement marking retroreflectivity data. There are tremendous opportunities in implementing the machine learning methods in areas that traditional methods are labor intensive and time consuming such as drawing embankment polygons using LiDAR DEM data.

Deep learning is a machine learning technique that allows computational models to learn the representation of massive and complex datasets without the need for explicit identification of prevalent features. Deep learning has recently been shown to outperform humans, in several fields such as computer vision and OpenAI's ChatGPT. MDOT SHA's massive amounts of engineering datasets as well as other data such as roadway and construction data have been accumulated over a long time period, which creates an excellent opportunity for establishing deep learning models to enable reliable prediction of engineering characteristics and other desired features from the massive datasets. There are many advantages of the machine learning-based approaches for field inspection/testing and imagery data modeling and prediction. Significant cost savings can be achieved by leveraging historical datasets and integrating with machine learning enabled work process. Recently machine learning algorithms have advanced rapidly; newer and more powerful machine learning models not available in previous applications and research in highway transportation fields are becoming available to address the challenging needs identified by OMT in design and construction, materials test data, management, operating optimization and field

inspection. If implemented, not only the material and condition characteristics of highway, and other transportation infrastructures can be estimated in the early phase of the project, but also scheduling/construction and maintenance can be optimized by data-driven decision making assistance enabled by accurate prediction with cutting edge machine learning methods and continual self-supervised learning of future incoming data.

While we are implementing machine learning into various areas in the MDOT SHA OMT, there are still many areas that would greatly benefit from these tools. This research is aimed to optimize and apply state-of-the-practice machine learning algorithms and application tools to address the emerging needs in the above-stated OMT needs for better operation and management of MDOT SHA highway assets. In Phase 2 research, we implemented recent advance in machine learning methods, specifically, machine learning models for LiDAR DEM data based landslide risk assessment, slope/embankment detection for geotechnical asset inventory, scarp line detection, concrete compressive strength test data modeling and prediction, pavement marking retroreflectivity data modeling and deterioration condition prediction.

RESEARCH OBJECTIVES

This project is aiming to investigate and optimize cutting-edge machine learning models for identified MDOT OMT datasets and derive previous invisible relationships and trends within the historical data. Machine learning models can be utilized by providing design engineers with historic data-based predictions. These predictions will be implemented to generate more accurate preliminary designs, which will result in significant fiscal savings. These machine learning methods will also help ease delays in material testing by giving data requesters information that can be immediately utilized. It is imperative that we get these machine learning models in place to expedite the process of providing better preliminary data to MDOT SHA customers, thus improving customer service, fiscal savings, and reduced delays across all areas of OMT. Data size and quality is an important issue to ensure machine learning model performance and will thus be studied for the newly identified data as well. These models can be used to assist with the decision-making process of MDOT SHA in project planning and construction.

Literature review will be performed for cutting edge machine learning algorithms optimal for newly identified datasets and application areas of interests to MDOT SHA. By leveraging on the progress made in Phase 1 project, we will identify more datasets where machine learning would be beneficial such as pavement marking retroreflectivity data. Different data formats including tabular datasets and images provided by MDOT SHA OMT office engineers will be investigated and processed (e.g., cleansing, filtering, filling missing values, feature identification and definition, imbalanced dataset remedy, etc.) for use by machine learning models.

Cutting edge machine learning models will be developed for object detection and instance segmentation in LiDAR (Light Detection and Ranging) DEM (Digital Elevation Model) data and other raster image datasets. Machine learning can be utilized by the various OMT Divisions for object detection to provide initial data sets for slope distress area, embankment, pavement markings, small structures, and other items catalogued from existing ARAN Imagery. Machine learning algorithms have advanced rapidly and thus tremendous opportunities exist here in implementing machine learning methods for object detection performance improvement.

Conduct case studies of additional highway datasets of interest to OMT including drilling data, geologic risk, concrete compressive strength test data and develop machine learning methods for these data. These machine learning models can be used and updated with new data to assist with the decision-making process of MDOT SHA in project planning and construction.

RESEARCH APPROACH

To achieve the objectives of this study, the following tasks were undertaken. This research considered the following seven types of datasets and corresponding machine learning algorithms for proper modeling of embedded data relations: LiDAR DEM data, raster tiff images, drilling data, geologic risk, pavement marking retro-reflectivity, concrete compressive strength test data. A brief description of each machine learning method is given below and more details on why they were chosen can be found in the corresponding chapters of this report.

Task 1. Project Management

The research team coordinated closely with MDOT SHA throughout the project in order to establish machine learning models for the selected highway datasets including machine learning models for LiDAR DEM data based landslide risk assessment (Task 2), instance segmentation based slope/embankment detection for geotechnical asset inventory (Task 3), instance segmentation based scarp line detection (Task 4), concrete compressive strength test data modeling and prediction (Task 5), pavement marking retroreflectivity data modeling and deterioration condition prediction (Task 6), and geotechnical drilling data (Task 7). In Tasks 5 to 7, oversampling method has been adopted to address the issue of data imbalance in the original datasets which results in increased accuracy in machine learning model prediction compared to previous results. Quarterly progress reports were prepared and submitted. Participation in project meetings coordinated by MDOT SHA with OMT staffs were attended regularly for machine learning model application needs and data requirements.

Task 2: LiDAR DEM data driven machine learning modeling of highway landslide risk assessment

The research team (here forth defined as UMD and MDOT team members) trained the highway landslide risk mapping and slope inventory. A geospatial machine learning model has been applied to assess the landslide risk in Maryland. Trained with historical slope failure data, the machine learning model includes LiDAR DEM data derived terrain feature variables and additional feature variables related to lithology, distance to roadways and streams. The model was applied to statewide 2-meter resolution LiDAR DEM data to generate the landslide risk map. Locations with risk score greater than 0.5 have been highlighted to inform engineers with increased landslide risk. Random forest model was trained to predict for landslide risk at a given location in Maryland.

Task 3: Instance segmentation machine learning model for embankment and slope polygon identification

The research team reviewed machine learning methods for detecting embankment polygons for geotechnical cut slope and embankment asset inventory. Instance segmentation machine learning model was trained to automatically identify and draw the embankment polygons in arc gis pro. Slope raster tiff images created from state-wide 2-m resolution LiDAR DEM data were used for

machine learning model training. Hyperparameter tuning of machine learning model such as training image size and training epoch numbers were performed to obtain the optimal model for embankment polygon identification. Validation of the machine learning identified embankment polygons were made by comparing with embankment inventory data provided by MDOT SHA. The research team also did literature review of recent publications in this field to ensure the adopted instance segmentation model has the state of art performance.

Task 4: Instance segmentation machine learning model for slope scarp identification

The research team developed and tested machine learning model for detecting scarp lines in arc gis pro. Instance segmentation machine learning model was trained to automatically identify and draw the scarp lines. Surface parameter slope raster tiff images created from state-wide 1-m resolution LiDAR DEM data were used for machine learning model training. Hyperparameter tuning of machine learning model such as training image size, image pixel depth and training epoch numbers were performed to obtain the optimal model for scarp line identification. Machine learning identified scarp lines were compared with the nventory scarp line data provided by MDOT SHA.

Task 5: Developing Machine Learning model for concrete strength test datasets

The research team trained neural network machine learning model for concrete compressive strength predictions by utilizing historical data from thousands of historic concrete strength tests with various mix design and concrete placement variables. Data cleanup was conducted to ensure that only the most reliable data was used in training a tabular neural network regression model. The model prediction yields a R^2 value equal to 0.87 when tested on validation data.

Task 6: Developing Machine Learning model for pavement marking retroreflectivity data

The research team also utilized Laserlux collected pavement marking retroreflectivity data to develop machine learning regression models and performed training for various marking materials prediction. Feature variables include county, route number, age, materials and AADT. Shapefile for the data frame was created for visualizing the geospatial distribution of pavement marking data in Arc gis pro. The use of pavement marking retroreflectivity machine learning model for condition deterioration model was also demonstrated.

Task 7: Developing machine learning model for geotechnical drilling data

The research team trained tabular data neural network models for various geotechnical drilling datasets, including locations, geology, elevation, major soil constituent, layer depth, groundwater table depth, SPT N-Value ranges, and bedrock depth. The resulting model is fully implemented and is utilized as part of the soil boring request process in MDOT SHA. The tabular data neural network model is current and has been updated for the drilling data based on parametric study of hyper-parameter values. Tuning the machine learning model for optimal hyper-parameter values was conducted. Comparison of the predicted results from Fast.ai neural network model and true value was also made by calculating the corresponding confusion matrices and plotting a scatter plot in this study.

Task 8: Final Report

The research team developed this final report that includes all deliverables and analyses as described in Tasks 1 to 7.

CHAPTER 2. LIDAR DEM DATA DRIVEN MACHINE LEARNING MODELING OF HIGHWAY LANDSLIDE RISK ASSESSMENT

The objective of this task was to train geospatial machine learning models to assess the landslide risk in Maryland. Random Forests machine learning algorithm is used to predict the probabilistic landslide occurrence likelihood over a large area along the road network in the state of Maryland. Trained with historical slope failure data, the machine learning model includes LiDAR DEM data derived terrain feature variables and additional feature variables related to geology, distance to route and cost distance to streams. The model was applied to statewide 2-meter resolution LiDAR DEM data to generate the Maryland landslide risk map. Locations with risk score greater than 0.5 have been highlighted to inform engineers with increased landslide risk.

In this study, the research team reviewed raster image data based Random Forest model training and predictions for landslide risk mapping. Random Forest classification machine learning model has been used to predict the probabilistic landslide mapping by Maxwell et al. (2020). Random Forest classifier is an ensemble tree-based learning algorithm that consists of many decision trees from randomly selected subset of training set. It aggregates the votes from an uncorrelated forest of decision trees to predict the class. Random Forest is thus a nonparametric, ensemble decision tree (DT) method capable of accepting continuous and categorical predictor variables to perform classification, regression and make probabilistic predictions. Random forest has many positive attributes for predictive modeling including its ability to accept a variety of input predictor variables that may be correlated and/or scaled divergently. Also, it is generally robust to complex feature space, can be trained quickly, can accept categorical predictor variables and can provide an assessment of variable importance based on the withheld or out-of-bag, data in each tree (Breiman 2001).

The random forest algorithm has been applied to many mapping and spatial predictive modeling tasks with remotely sensed and/or geospatial data as input (e.g. Youssef et al. 2016). It has also been used extensively to obtain probabilistic predictions as opposed to classification products. Maxwell et al. (2020) recently implemented the Random Forest algorithm for predicting the

probabilistic mapping of landslide occurrence or susceptibility in West Virginia. This Random Forest modeling approach was also adopted for this study of landslide risk assessment in Maryland. The initial training data consisting of a total of 1,798 slope failure samples was provided by the West Virginia University team listed in the reference by Maxwell et al. (2020), in which the details of their model can be found. Through integration of machine learning and geospatial predictive modeling, landslide probability maps are generated in this study to visualize where slope failures are likely to occur along the Maryland highway routes.

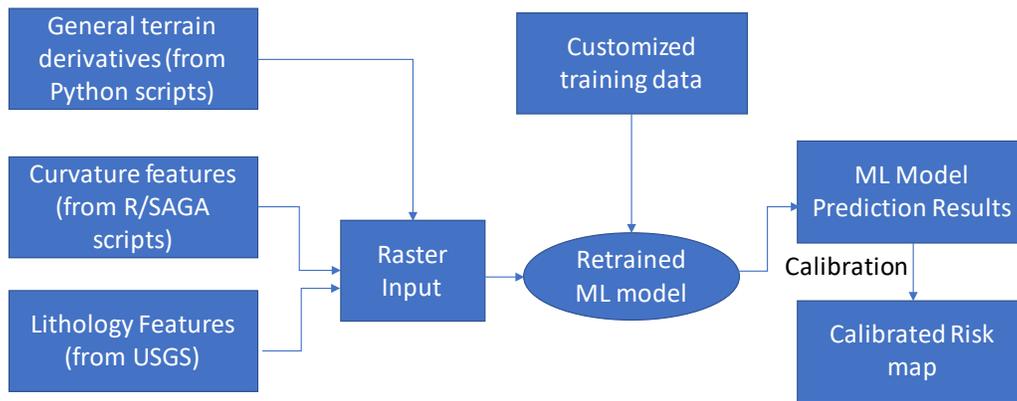


Figure 1. Framework of raster-image based landslide risk machine learning model

LiDAR (Light Detection and Ranging) is an active remote sensing method that relies on laser range finding. Using the two-way travel time of reflected laser pulses detected by the sensor, GPS (global position system) locations, and aircraft orientation and motion from an inertial measurement unit, horizontal and elevation coordinates of the reflecting surface can be estimated at a high spatial resolution. Further, a single laser pulse can potentially result in multiple returns, allowing for vegetation canopy penetration and the mapping of subcanopy terrain, in contrast to other elevation mapping methods such as photogrammetry method (Maxwell et al. 2020).

LiDAR data have been applied to a variety of terrain mapping and analysis tasks. For example, many studies have investigated the mapping of slope failures, such as landslides, using terrain variables derived from LiDAR (e.g., Ardizzone et al. 2007, Jaboyedoff et al 2012, Van Den Eeckhaut et al 2012). Another common application is modeling the likelihood of slope failure occurrence or landslide risk (Maxwell et al. 2020). In a review of the use of LiDAR in landslide investigations, Jaboyedoff et al. (2012) suggest that LiDAR is an essential tool for landslide risk management and that there is a need to develop methods to extract useful information from such

data. In this study, LiDAR DEM data has been used to generate a variety of terrain feature variables for the landslide risk machine learning model. DEM (digital elevation model) is a digital representation of the earth's topographic surface in which surface features like buildings and vegetation have been removed. A high-resolution DEM can be derived from LiDAR point-cloud data by stripping away the surface features and sampling the ground elevation in uniform increments to produce a bare earth model (USGS).

Previous research works (Goetz et al. 2015; Maxwell et al. 2020) suggest that terrain variables are highly important in predicting landslide occurrence and susceptibility. Goetz et al. (2015) also suggest that empirical or trained models that incorporate terrain variables often outperform physical models that attempt to model slope failure susceptibility based on physics based understanding of the slope stability problem.

The current machine learning model adopted here for landslide risk assessment incorporates a total of 35 features (among which 23 features generated from Python script in arc gis pro, 9 features generated from R script in R geospatial analysis package, and 3 feature created from arc gis pro). The following 23 terrain variables are derived as the outputs from Python scripts in arc gis pro: Slope Gradient (slp), Slope Position (sp 3), Topographic Roughness (tr or rph 3), Topographic Dissection (td or diss 3), Mean Slope Gradient (SlpMn 3), Site Exposure Index (sei), Heat Load Index (hli), Linear Aspect (LnAsp), Surface Relief Ratio (srr 3), Surface Area Ratio (sar), Profile Curvature (PrC 3). Those features with 3 in the parenthesis are further divided into three individual features with different window sizes for averaging purpose. The following additional nine terrain variables are derived from R package and also used for the landslide risk machine learning model: Plan Curvature (Plc 3), Cross-Sectional Curvature (csc 3), Longitudinal Curvature (LnC 3). The importance ranking of the corresponding feature variable to the landslide risk assessment model are shown as the number in the parenthesis: slp (1), sp1 (21), sp2 (27), sp3 (32), rph1 (tr1, 7), rph2 (tr2, 23), rph3 (tr3, 22), diss1 (td1, 6), diss2 (td2, 24), diss3 (td3, 34), slpmn1 (12), slpmn2 (16), slpmn3 (18), sei (13), hli (42), asp_lin (LnAsp, 11), srr1 (4), srr2 (28), srr3 (36), sar (2), proCrv1 (PrC, 10), proCrv2 (30), proCrv3 (20), tanCrv1 (csc, 3), tanCrv2 (15), tanCrv3 (35). The number after the letters in the abbreviated feature variable name refers to the window size for averaging the corresponding feature variable.

For example, sp1, sp2, and sp3 has the average slope position value based on a window size of 7x7, 11x11 and 21x21 cells respectively. Each cell in the raster image measures 2 m x 2m since 2-m resolution LiDAR DEM data was used for this task.

As demonstrated in the study by Maxwell et al. (2020), the most important five variables in the model are topographic slope (Slp), surface area ratio (SAR), cross-sectional curvature (CSC), surface relief ratio (SRR) and plan curvature (PIC). In addition to the terrain feature variables, the machine learning model in this study also considers non-terrain variables, such as variables relating to Geomorphic Presentation or lithology (lith, 17), distance to nearest road (USD), and cost distance to streams (StrmC, 19).

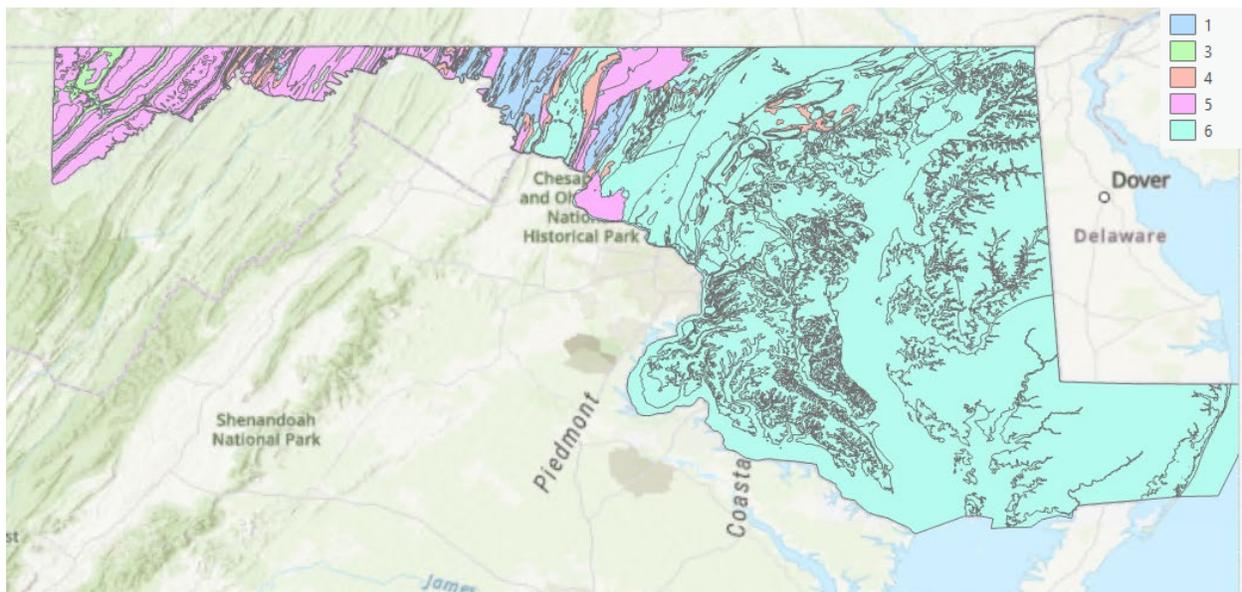


Figure 2. Lithology map of Maryland (1= low relief carbonates, 2 = low relief mudrock, 3= major ridge formers, 4= Moderate or variable quartzose ridge formers, 5= Moderate relief clastic rocks, 6= others, 7= Shaley units with interbedded sandstone)

The lithology map for Maryland was downloaded from the USGS website (www.usgs.gov) and saved into raster tiff files. Since there are 34 different rock types present in the Maryland lithology map downloaded from the USGS lithology data, a procedure in arcgis pro was developed by the UMD team to generate the statewide lithology map raster tiff data compatible with the lithology classification scheme in the original model. The lithology map used for this study is shown in Figure 2. ‘Distance to nearest roads’ feature variable has also been included in the current landslide risk machine learning model. Tile raster files have been extracted from

Maryland highway routes shape file. Euclidean Distance feature layer were created by Euclidean distance function. A sample cost distance to nearest stream map is shown in Figure 3 for Anne Arundel County.



Figure 3. Map of cost distance to streams of Anne Arundel County

Generated machine learning model required features input raster data for each county in Maryland, and established baseline estimation of computing and storage resource requirements for statewide landslide risk map production. To compute the features, the running time for R script ranges from 40 min to 1 h for each tile, the running time for all 57 tiles in Washington (WA) County is estimated to be 50 hrs. The trained model provides a f1-score of 0.86 for each landslide risk/non-landslide risk category (20% data was used for the validation test). The prediction results for each tile is visualized in the raster image of landslide risk map, as shown in the developed landslide risk maps for WA, AL and GA county in Figure 4 below.

Table 1. Computing information for Anne Arundel County landslide risk map generation

ML model feature variables	Raster Data size	Computing time (hours)	Software Package
Python Features	61.3 GB	15	Arcgis pro toolbox, ArcPy
R Features	15.6 GB	4	R scripts for geospatial analysis (SAGA, spatialEco package)
Lithology	30.2 MB	-	Arcgis pro
Distance to Road	1.82 GB	-	Arcgis pro
Cost Distance to Stream	1.12 GB	-	Arcgis pro

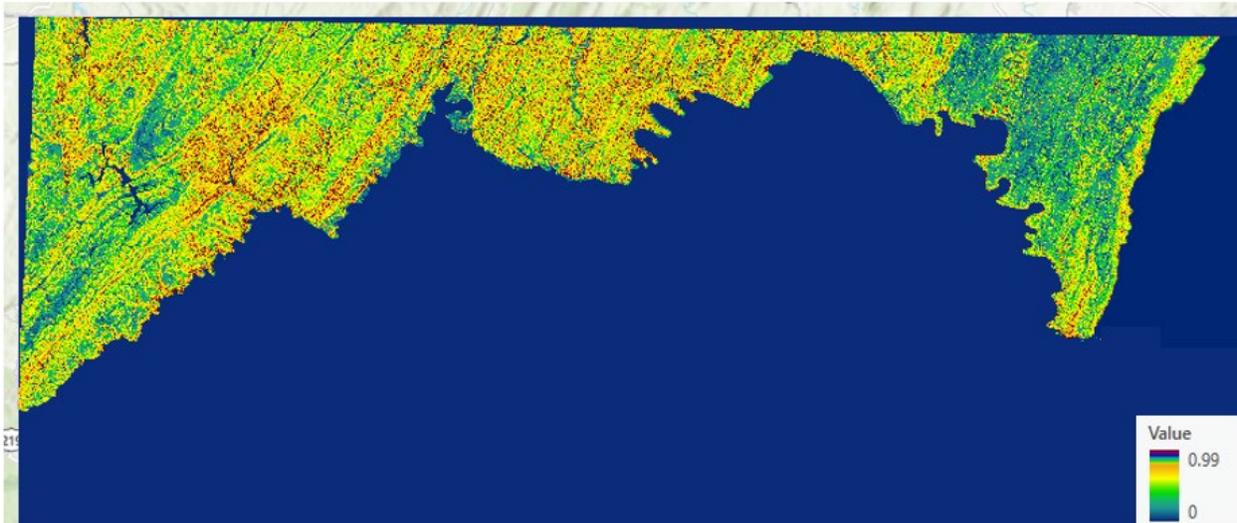


Figure 4. landslide susceptibility map of western four counties (GA, AL, WA) in Maryland

Steps to generate the risk map are listed as follows: 1. The DEM raster tiff file for the considered county was extracted from the statewide 2-m resolution LiDAR DEM file (elevation unit is foot). 2. Computing the geographical features in Arc gis pro, R studio and Jupyter notebook. 3. Train the Random Forest machine learning model for landslide risk assessment in Jupyter notebook. 4. Predict the confidence score by Py spatial machine learning module in Jupyter notebook. Table 1 above summarizes the computing time and data storage requirements for three major steps of Anne Arundel (AA) County landslide risk map generation based on a Dell Optiplex 9020 desktop computer with 3.6GHz Intel Core i7-4790 and CPU memory of 32 GBytes. The original

R code was updated by removing unused codes to shorten the computational time. ArcGIS pro 2.7 version was used.

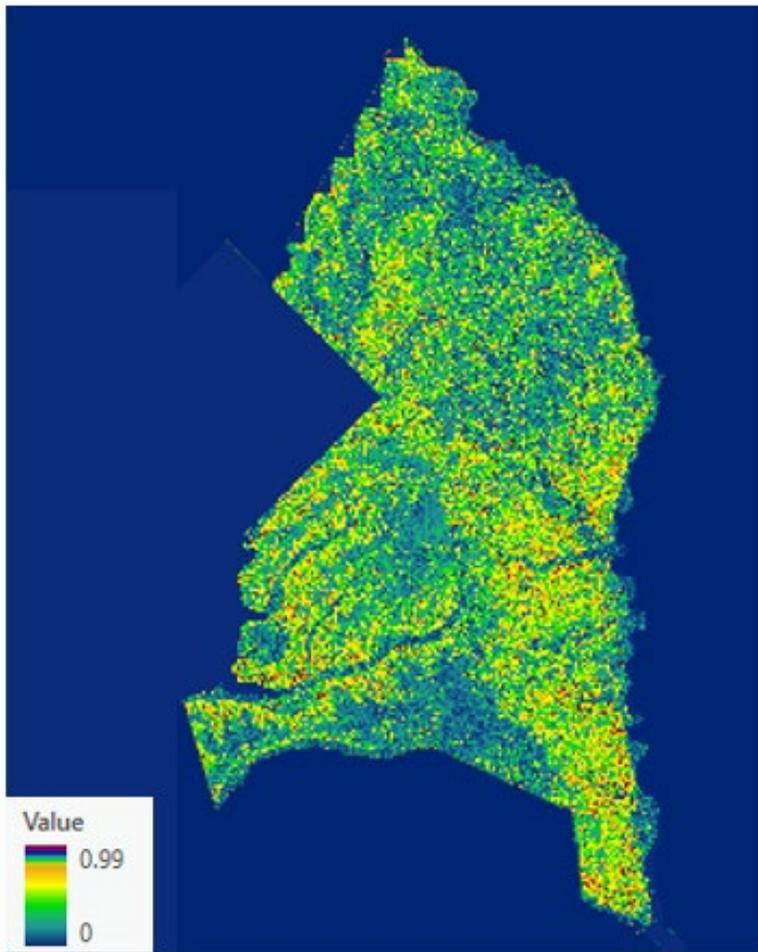


Figure 5. Machine learning generated landslide risk map of Prince George County

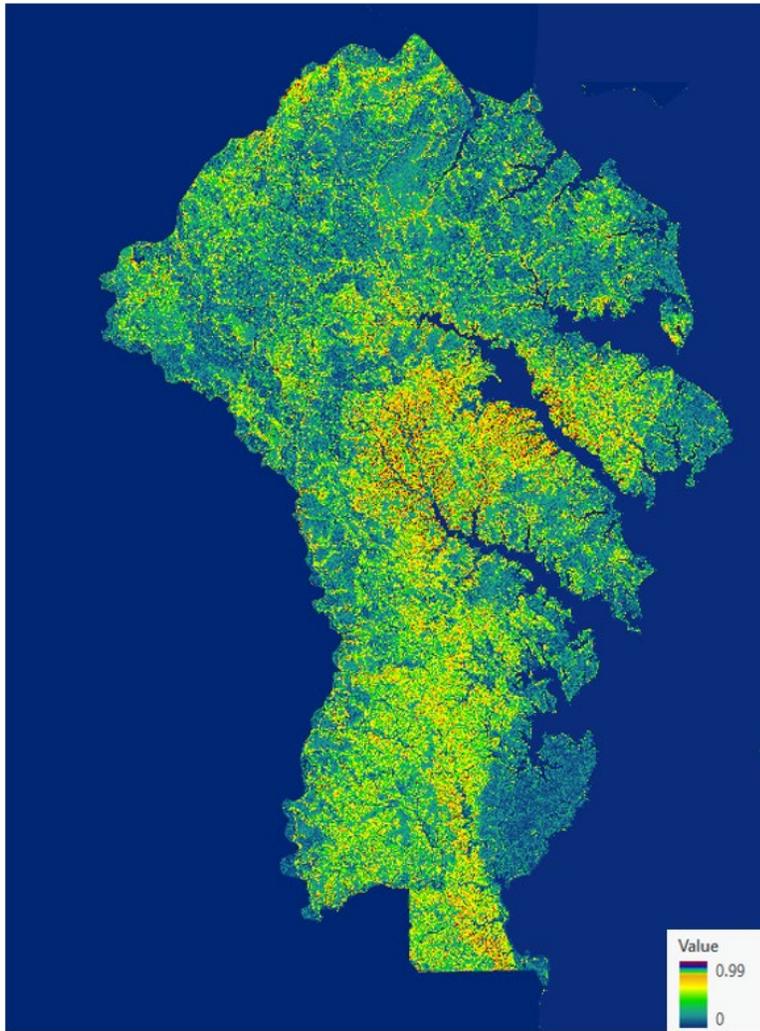


Figure 6. Machine learning generated landslide risk map of Anne Arundel County

CHAPTER 3. INSTANCE SEGMENTATION MACHINE LEARNING MODEL FOR EMBANKMENT AND SLOPE POLYGON IDENTIFICATION

The objective of this task was to train machine learning models for automated detection of highway embankments and corresponding polygon generation using raster image instance segmentation method. Literature review of image object detection methods was first carried out for embankment polygon generation to identify the cutting-edge machine learning algorithm and software tool for this task. Statewide LiDAR DEM data was used to for this purpose.

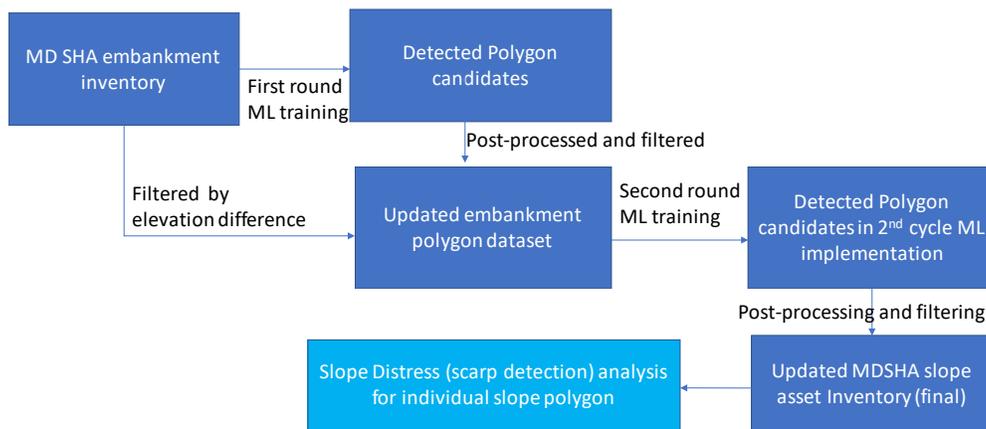


Figure 7. Flowchart of embankment and slope polygon detection using machine learning method

The instance segmentation tool in arc gis pro was used to develop the machine learning (specifically instance segmentation method) model for highway embankment polygon identification. Additional Python scripts were developed to calculate certain features (e.g., slope sectional height difference from transect lines, area, etc) and apply composite machine learning models (by combining tabular data models) to post-process the detected embankment polygon datasets such as removing those embankments with projected area smaller than a given value (e.g., 1000 square foot). Other post-processing works of the identified polygons include generating transect lines along the Maryland highway (IS, US, MD routes) road network centerlines and developing python scripts to calculate the maximum sectional height difference for each intersected transect line segment and the corresponding embankment polygon, as well as generating embankment polygon attribute table for summary. Fine-tuned instance segmentation machine learning models were developed in second cycle with those embankment polygons

obtained and post-processed from the first-cycle implementation of the machine learning model, as shown in Figure 7.

Deep learning is a machine learning technique that allows computational models to learn representation of massive and complex datasets without the need for explicit identification of prevalent features. Deep learning has recently been shown to outperform humans, in several fields such as computer vision and pattern recognition (Bengio et al. 2021). Combination of deep learning with high-resolution LiDAR DEM data creates an excellent opportunity for establishing machine learning models to automate and enable reliable prediction of surface features such as highway embankment and scarp distress areas in cut slopes. Instance segmentation can be viewed as a combination of semantic segmentation and object detection. Whereas an object detection system coarsely localizes multiple objects with bounding boxes, Instance segmentation produces a segment map of each category as well as each instance of a particular class, therefore, providing a more meaningful inference on an image. Object detection involves identifying the presence, location, and type of one or more specified objects in a given picture or video images. It builds upon methods for object recognition, object localization, and object classification. In recent years, deep learning techniques are achieving state-of-the-art results for object detection, such as on standard benchmark datasets and in computer vision competitions. Notable is the “You Only Look Once,” or YOLO, family of Convolutional Neural Networks that achieve near state-of-the-art results with a single end-to-end model that can perform object detection in real-time (Redmon and Farhadi 2018).

For object detection algorithm, the whole system can be divided into two major components: Feature Extractor and Detector; both are multi-scale. When a new image is loaded for object detection, it goes through the feature extractor first so that we can obtain feature embeddings at three (or more) different scales. These features are fed into three (or more) branches of the detector to get bounding boxes and class information. Inspired by ResNet and FPN (Feature-Pyramid Network) architectures, the feature extractor of Yolo v3 uses is Darknet-53, which contains skip connections (like ResNet) and 3 prediction heads (like FPN); each processing the image at a different spatial compression (Redmon and Farhadi 2018). ResNet brought the idea of skip connections to help the activations to propagate through deeper layers without gradient

diminishing (He et al. 2015). Darknet-53 borrows this idea and successfully extends the network to 53 layers. In object detection, a detection head follows after this feature extractor. Convolutional neural network (CNN) is also used in the object detection algorithm. The convolution/pooling layers before the final densely connected neural network, are there to extract and abstract patterns from the input data, in another word, engineer features that are useful for the prediction task. However, in more challenging dense prediction tasks such as pixel-level object detection or long-term prediction, both global and local information are needed for accurate prediction, requiring larger receptive fields. Two naive approaches to expand the receptive field are to deepen the network and to enlarge the filters (Redmon and Farhadi 2018). The former introduces computational and optimization difficulties due to problems of exploding vanishing gradients, hyper-parameters tuning, and memory explosion. The latter drastically increases the model complexity, making the already expensive training on dense prediction more challenging.



Figure 8. Slope composite band raster image of Washington County

This study combines the Mask R-CNN and LiDAR for detecting the embankment features from slope raster imagery data as shown in Figure 8. Specifically the Mask R-CNN method is

implemented. This method is an extension of faster R-CNN, which allows for convolution to be applied on regions of the image as opposed to the entire scene. This involves generating convolution feature maps that are then applied to individual subsets of the image, called regions of interest (RoI), defined by the region proposal network (RPN) (Maxwell et al. 2020). Large training datasets including a variety of embankment and slope features for calibrating and validating machine learning methods have been created in the first cycle of training, which were then used for the second round of machine learning model training in this study. Region-based Convolutional Neural Network (RCNN) model was utilized to detect the slopes. This method is a deep learning instance segmentation technique that performs pixel-level segmentation on detected objects. Since 2-m resolution LiDAR DEM data was used to create the slope raster images, each cell (or pixel) of the raster tiff image measures 2m x 2m in space. This instance segmentation model has been trained for 3-categories classification so that embankment, soil cut slope and rock cut slope can be explicitly identified. ResNet with 50 layers was used for the instance segmentation model.

The first round embankment detection machine learning model used training chips generated from the original embankment inventory data provided by MDOT SHA. A total of 11,884 training chips were generated for the consolidated dataset. Each training chip size is 256x256, which corresponds to 512m x 512m in space since 2-m resolution LiDAR DEM data was used to create the slope raster images. A sample training chip is shown in Figure 9 below. For the embankment polygons identification in Arc gis pro environment using the machine learning method, the training dataset was augmented by appending the initial training chips dataset with 5,000 rotated training chips (45-degree data augmentation) from the three western counties (GA, AL, WA) in Maryland and re-trained embankment detection model with this expanded training dataset. The consideration for choosing the three western counties only for rotation augmentation are two: (1) those highway roads in the region run mostly in east-west direction so rotation augmentation of the corresponding embankment and slope polygons would provide helpful information in training machine learning model; (2) there are many slope polygons in this region due to its mountainous terrain feature and adding these polygons in rotated orientation would also help with the qualify of the entire training dataset.



Figure 9. Sample training chip with a 256x256 cell size for embankment detection model

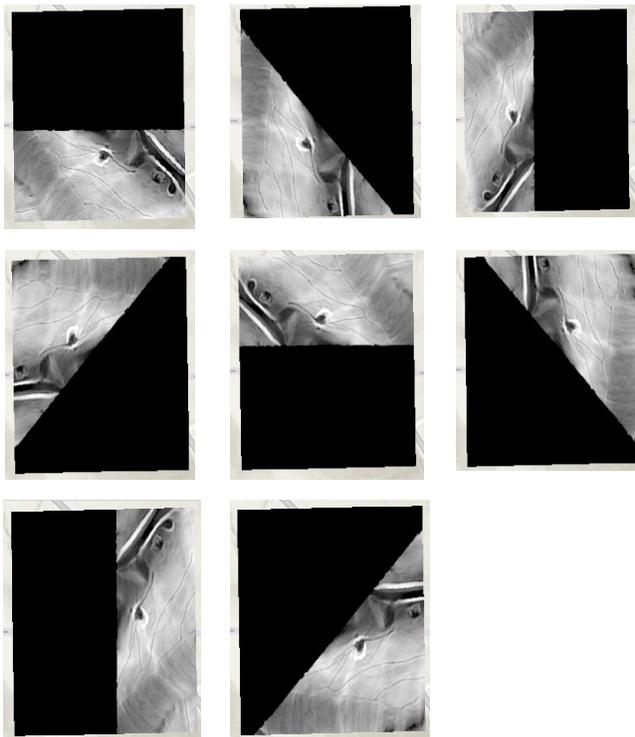


Figure 10. Embankment polygon mapping with training chip rotation augmentation

This machine learning method can automatically detect and draw the outlines of desired embankment polygons with regular shapes and smooth edges as shown in Figure 11. The

embankment detection procedure includes the following steps: 1. Importing the raster file of the training region; 2. Importing the vector shape file of the training polygons; 3. Importing the vector shape file of the mask of the training region (e.g., Washington county border); 4. Creating slope shade raster of the raster file using the “slope” tool in Arc gis pro; 5. Converting the raster to 8-bit unsigned format using the “copy raster” tool in Arc gis pro; 6. Creating a 3-band image from the converted raster; 7. Creating the training images using the “export training data for deep learning” tool in Arc gis pro. 8. Train the machine learning model for embankment polygon detection using the “train deep learning model” tool in Arc gis pro. Typical epoch number is 20 to 30 based on trial and error test of this case study. A sample training loss curve is shown in Figure 12. Batch size is set to be 8. 9. Detecting the objects in the study area using the “detect object using deep learning” tool in Arc gis pro. Threshold confidence score value can be adjusted to generate more embankment polygons. For example, by decreasing the threshold to 0.5, more polygons are detected at some locations, as shown in Figure 13.

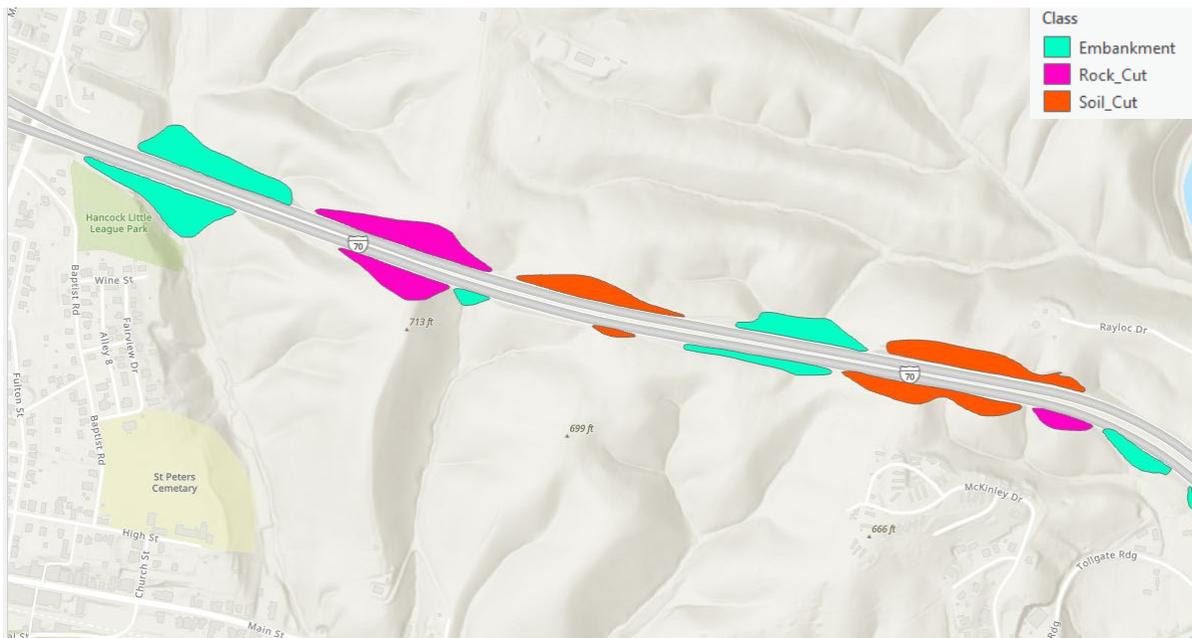


Figure 11. Embankment polygons detected using the Instance Segmentation method at a location in Washington County along Highway I-70

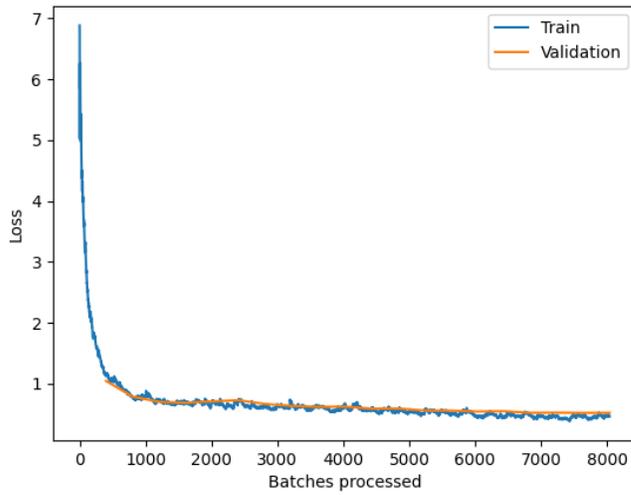


Figure 12. Sample training loss curve for the embankment detection machine learning model

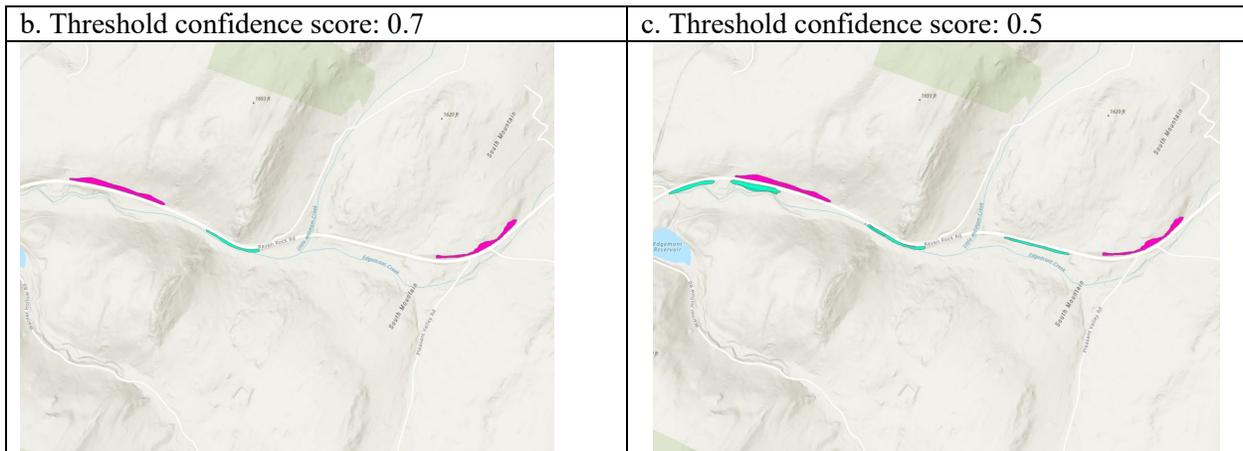
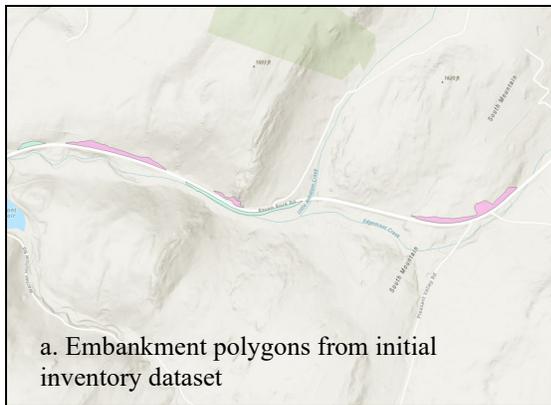


Figure 13. Comparison of machine learning generated embankment polygons with inventory data

The detected polygons from the first round training model predictions were filtered by applying a minimum requirement of 10-ft sectional height (elevation) difference, 1000 sq. ft. minimum (projected horizontal plane) area and 0.6 confidence score output for the machine learning model prediction. The detected embankment/slope polygons were further post-processed after filtering, including applying dissolve tool, multipart to single part break down, non-gap union tools. Table 1 and Figure 14 show the amount and distribution of filtered polygons from first round machine learning model detected embankment and slope polygons.

Table 2. Number of filtered polygons from first-cycle machine learning prediction appended to MDOT SHA inventory dataset to make the slope-type dataset more balanced for training

	Initial Inventory Data	First-cycle ML Prediction (Confidence score > 0.6)	Added Samples with adjusted passing confidence score	Final balanced dataset
Embankment	758	2758	538 (CS ≥ 0.92)	1296
Soil cut slope	342	931	520 (CS ≥ 0.75)	862
Rock cut slope	531	241	241 (CS ≥ 0.6)	772

Note: ML = machine learning

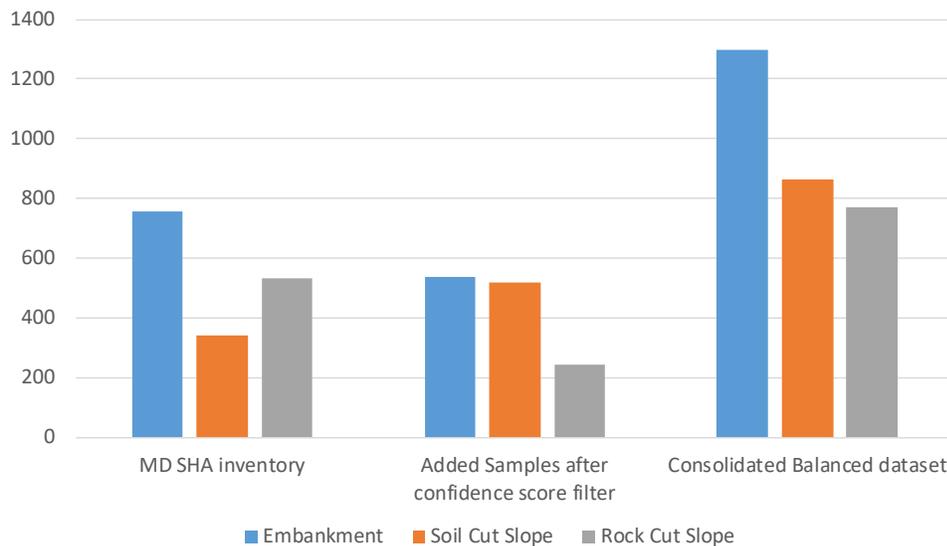
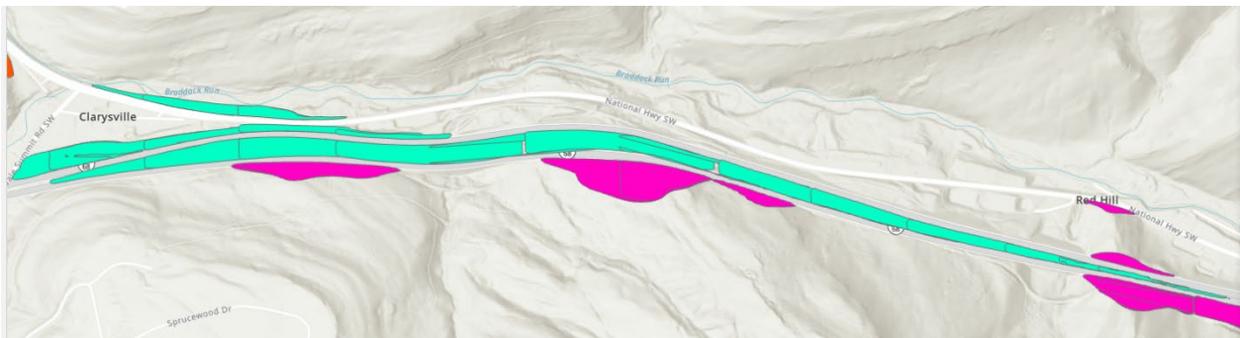
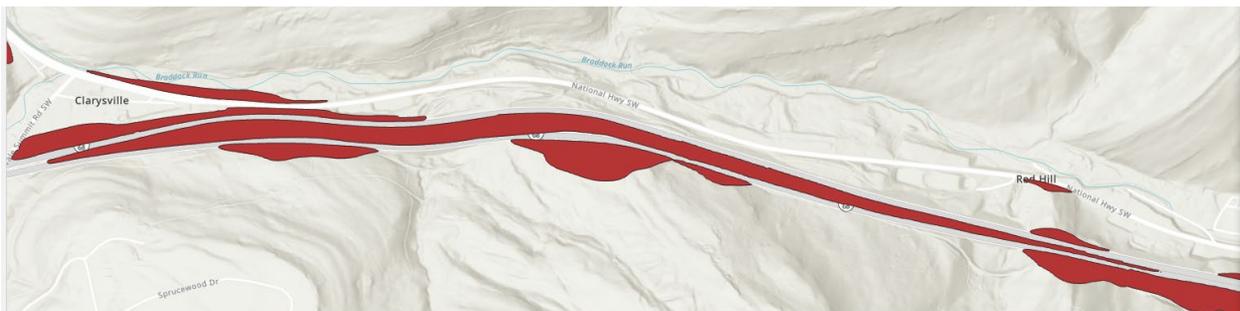


Figure 14. Histogram of the filtered polygons from first-cycle machine learning detected embankment and slope polygons

Post-processing of machine learning model detected embankment polygons is critical to creating a high-quality training dataset that can be used for second round machine learning model training. The polygons drawn by instance segmentation are further filtered by applying confidence score threshold (≥ 0.6) and transect line based elevation difference (≥ 10 ft). Dissolve and non-gap union tools were applied to candidate polygons to remove the overlapped polygons and fill void and holes of the polygons in arc gis pro. Multipart to single part tool was also applied to separate the polygons and spatial join was applied to attach attribute features for the processed polygons. Area filter (> 1000 sq. ft) was applied to remove small non-embankment polygons. Dissolve tool is applied to connect polygons for the same embankment. Figure 15 below shows the embankment polygons before and after applying these post-processing steps.



(a)



(b)

Figure 15. Comparison of embankment polygon (a) before post-processing; (b) after post-processing

polygons were identified and added to the new training dataset for next phase machine learning model training. It is worth noting that the total number of embankment polygons selected for training can be further increased if needed by lowering the confidence score threshold so more polygons can be included. The updated embankment polygon dataset for second-cycle machine learning model training was then used for retraining and the retrained machine learning model was applied again to detect embankment with the 2-m resolution statewide LiDAR DEM dataset.

In order to calculate the sectional height difference and mile-point labeling for the corresponding embankment, the intersection line segment between the 1/1000 mile transect lines and identified embankment polygons was generated in arc gis pro, as shown in Figure 16. The maximum and minimum elevation along each transect line segment (i.e. only the segment inside the embankment polygon) were then determined from the LiDAR DEM raster images by using the zonal statistics tool in arc gis pro. The shapefile of Maryland 1/1000 milepoint marker data was used for this purpose. 1-ft and 1000-ft long transect lines are generated along the Maryland road networks (including all IS, US, MD and RP routes). Figure 16 shows the mile point markers attached to 1-ft transect lines and joining the 1-ft transect line attributes with 1000-ft transect lines.

Additional python scripts were developed in this study to calculate several desired features (e.g., projected area of embankment, bridge linear intersection, etc.) for the highway embankment polygons identified from instance segmentation model using LiDAR DEM data and combined tabular machine learning models for embankment polygon filtering. For the embankment polygons identification in Arc gis pro, machine learning derived embankment polygon intersection with Maryland bridge linear was performed, and a polygon cutting method was developed to slice apart the polygons at bridge. Embankment polygon datasets were further processed by filtering out those unwanted polygons (e.g., near river banks) and applying bridge linear slicing to remove the embankment polygon portion under bridge deck based on the embankment polygons detected from 2-m resolution LiDAR DEM data.

Table 3. Comparison of instance segmentation machine learning model detection results (confidence score ≥ 0.4)

	ML Model w/o 3 county rotation	ML Model w/ 3 county rotation	ML Model w/ 5000 images from 3 county rotation
# of Embankment polygons	46500	18593	55044
# of Rock cut slope polygons	13612	5096	13444
# of Soil cut slope polygons	9208	5300	8453

ML = machine learning

A 45-degree rotation augmentation was applied to three western counties (GA, AL and WA) polygons to improve machine learning model performance. Training chip inventory for polygons with such rotation augmentation has 25,484 samples and is larger than the original training dataset. Two new machine learning models were created: one trained with all rotation augmentation data, the other one trained with 5000 training chips randomly picked from all rotation augmentation raster chips. The instance segmentation machine learning model was trained with the original pre-trained weights. The model with 5,000 rotation chips has an average precision score over 0.4 for each category. Figure 17 shows samples of instance segmentation machine learning model detected embankment and slope polygons with and without rotation training chips.

According to the machine learning model detection quantity results in Table 4, the instance segmentation model with additional 5000 rotation augmentation training chips can detect more embankment polygon candidates than the other two instance segmentation models. There are 25,123 polygons left after applying the dissolve and union tool, filtered by only keeping those polygons with a 1200ft-wide buffer zone along the highway centerline. The filtering confidence score threshold for this machine learning model detection is set to be 0.4, therefore, only those embankment or slope polygons with confidence score ≥ 0.4 are kept. Additional filtering criteria include the following: Maximum sectional elevation difference of the embankment polygon ≥ 10 ft; Embankment Projected Area ≥ 1000 . By applying the elevation difference filter along the transect lines using the 1/1000th mile marker shapefile: ≥ 10 ft filter yields 1,650 polygons; ≥ 15 ft filter yields 1,558 polygons. Comparing with the 1/100 mile maker intersect line: ≥ 10 ft filter

Figure 17. Comparison of machine learning model detection embankment/slope polygon samples: (a) w/ 5000 chips randomly selected from three western counties (GA, AL and WA) rotation augmentation; (b) w/ all rotation augmentation training chips for three western counties; (c) without rotation augmentation chips

The embankment candidate polygons are intersected with the Maryland bridge linear features, those polygons will be cut by straight line perpendicular to the bridge linear line at the intersection edges, as shown in Figure 18. It was found that only 220 out of 11,038 detected embankment polygons intersected with the Maryland bridge linear features. The embankment candidate polygons were sliced and divided by the bridge linear features with buffer zone of its width equal to the bridge deck width given in the Maryland bridge linear dataset. Sample embankment polygons sliced apart by bridge linear feature are shown in Figure 18.



Figure 18. Slicing machine learning model detected embankment polygon with bridge linear feature: (a) Polygon before cutting by bridge buffer polygon; (b) Polygons after cutting

Detection on the 2-m resolution statewide LiDAR DEM for first cycle model with the updated inventory dataset was performed with a total computing time of 17 hours 45 minutes. Total

polygons detected from first cycle machine learning model detection are 69,320 and the number of detected polygons with confidence score over 0.6 is 34,999; Additional polygon candidates were detected from applying the second round embankment detection machine learning model to the updated training dataset. Figure 19 compares the embankment polygons detected by second round machine learning model with those from first cycle machine learning model detections.

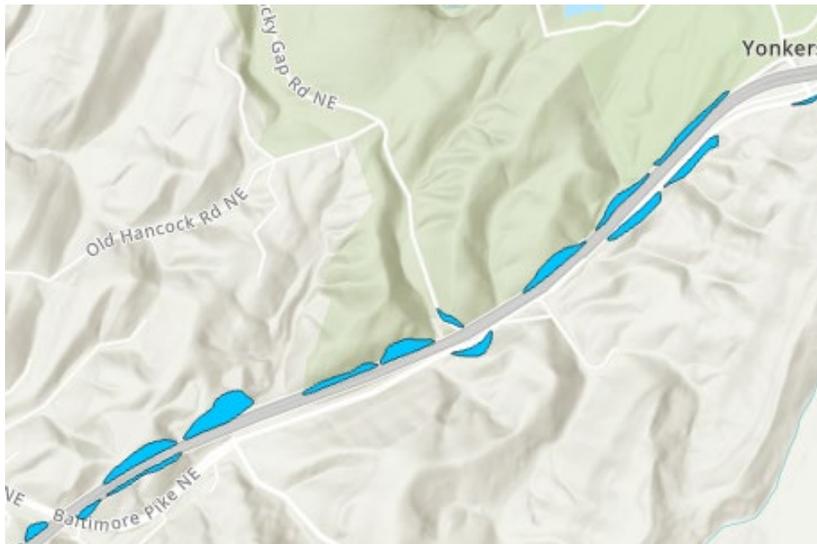


(a)

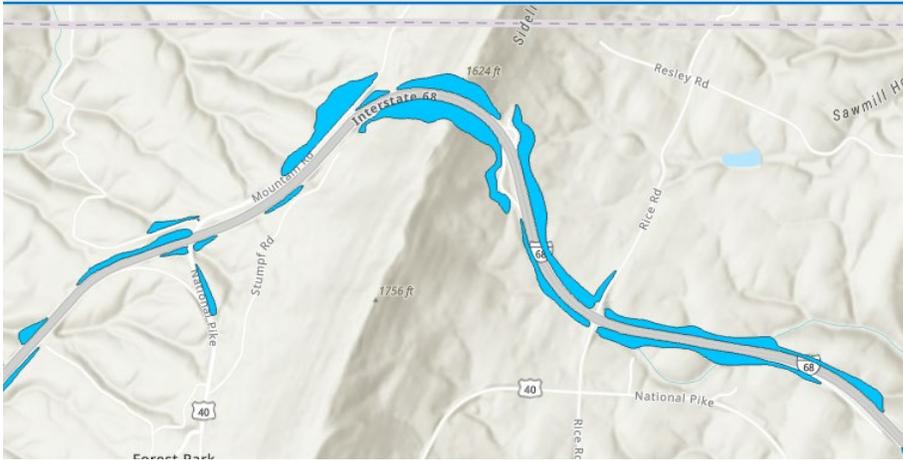


(b)

Figure 19. Embankment Detection by second round ML model: (a) detected polygons in 2nd round ML model; (b) training dataset from 1st round ML prediction



(a)



(b)
Figure 20. Sample embankment polygons detected from 2nd round machine learning model

FID	Shape	Class	Confidence	Shape_Leng	Shape_Area	ROUTEID	MPSTART	MPEND	CAND_ID	
1	0	Polygon	Embankment	93.026447	972.371357	17093.021369	19000MD00667	01EE*...	13.108 13.195	19000MD00667 01EE*****013107013194
2	1	Polygon	Embankment	80.582649	2727.537765	43249.219057	19000US00013	095S*...	19.452 19.671	19000US00013 095S*****019451019670
3	2	Polygon	Embankment	80.582649	2727.537765	13906.916626	19000US00013	095S*...	19.816999 19.885	19000US00013 095S*****019816019885
4	3	Polygon	Embankment	89.625847	2167.826101	45811.066787	23000US00013	095S*...	0.158 0.362	23000US00013 095S*****000158000362
5	4	Polygon	Embankment	94.315541	1832.717061	46478.379262	19000US00013	01NN*...	0.439 0.654	19000US00013 01NN*****000439000654
6	5	Polygon	Embankment	82.556027	1555.164197	19891.653154	23000US00013	01NN*...	5.972 6.136	23000US00013 01NN*****005972006136
7	6	Polygon	Embankment	97.529662	2811.831707	51085.325007	19000US00013	095S*...	19.18 19.42	19000US00013 095S*****019180019420
8	7	Polygon	Embankment	88.315833	1957.567583	45681.85308	19000US00013	01NN*...	0.67 0.854	19000US00013 01NN*****000670000854
9	8	Polygon	Embankment	43.570551	817.100116	11151.366586	19000US00013	095S*...	17.811001 17.905001	19000US00013 095S*****017811017905
10	9	Polygon	Embankment	50.638896	844.13589	14746.498286	19000US00013	01NN*...	2.38 2.468	19000US00013 01NN*****002380002467
11	10	Polygon	Embankment	90.522885	2922.730889	93129.022452	19000MD00361	01EE*...	4.208 4.332	19000MD00361 01EE*****004208004331
12	11	Polygon	Embankment	96.170163	1243.24883	20365.237049	18000MD00235	01NN*...	0.509 0.558	18000MD00235 01NN*****000509000558

Final_Score	Recommendation	CountyID	County	District	Route_Type	Route_Number	Year_Built	Construction_Contract_No	Construction_FMIS	AADT	AASI
						19000MD00667-1-----					
						19000US00013-2-----					
						19000US00013-2-----					
						23000US00013-2-----					
						19000US00013-1-----					
						...					
						01000CO00591-1-----					

Figure 21. Machine learning model detected embankment polygon post-processing: Attaching 1/1000th mile marker milepoints attribute

For the embankment polygons identification in Arc gis environment using machine learning method, UMD team manually checked the machine learning predicted embankment polygon inventory with confidence threshold of 0.4, attached the route ID and starting and ending 1/1000th milepoints feature to the detected embankment/slope polygon. The attribute fields corresponding to the inventory data were added to the detected polygon attribute table. The route ID format in the detected embankment polygon attribute table were updated and thus route_ID has been updated with the new criteria as shown in Table 4.

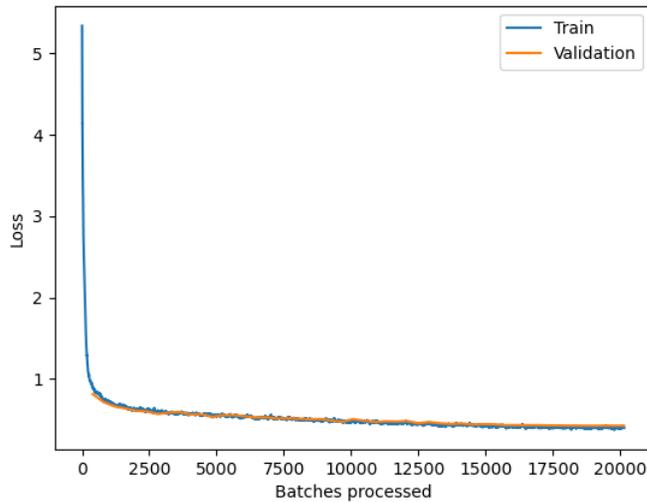


Figure 22. Training loss curve from training embankment detection machine learning model using the 1-m resolution Washington County LiDAR DEM data

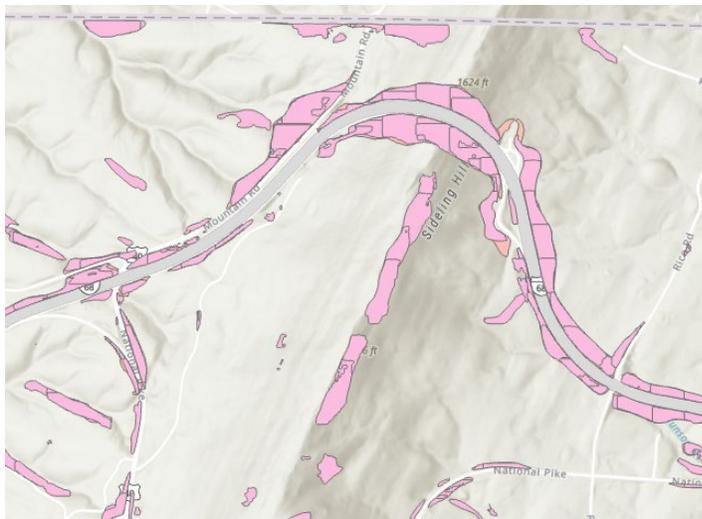


Figure 23. Sample machine learning detected embankment and slope polygons using 1-m resolution LiDAR DEM data at Washington County

Additionally, UMD team also tested a new embankment detection machine learning model training and prediction with 1-m resolution Washington County LiDAR DEM data. Two different training chip sizes (256 x 256 cells, and 512 x 512 cells). The 512x512 training chips were considered to match the same spatial dimension of those 256x256 training chips created from 2-m resolution LiDAR DEM data. It was found that the 1-m resolution LiDAR DEM data contains too many local features and thus machine learning model trained from this higher resolution LiDAR DEM data (compared with 2-m resolution DEM data) also captures undesired patches in the mountain area not related to embankments. Due to increased spatial resolution associated with the 1-m resolution LiDAR DEM data, computing time and resources requirements (e.g., GPU memory requirements) are significantly increased. Therefore, it was determined that for embankment/slope detection, using 2-m resolution LiDAR DEM is sufficient and should be adopted for this task.

MDOT SHA conducted a manual review of all slopes to refine the resulting polygons to those most relevant to highway asset management and risk analysis. Several polygons were eliminated where they do not relate to highways maintained by MDOT SHA or are not on the National Highway System. Many of the removed polygons were identified related to slopes for railroads, county-maintained highways, urban retaining walls, and other property within the analysis buffer but not related to highways. Ultimately, this analysis assisted MDOT SHA with adding 8,627 new slopes to the geotechnical asset management inventory, comprised of 5,727 additional embankments, 505 Additional Rock Cuts Slopes and 2,395 additional soil cut slopes.

CHAPTER 4: INSTANCE SEGMENTATION MACHINE LEARNING MODEL FOR SLOPE SCARP IDENTIFICATION

The research team also investigated the image instance segmentation method for slope scarp line mapping and machine learning-enabled automated scarp polylines generation in arc gis pro. A higher (1-m) resolution statewide LiDAR DEM data was used for this task because high spatial gradient value typical of scarp distress area. Surface parameter slope tiff raster images were generated from the 1-m resolution LiDAR DEM data for the area and the training chips with 3-m wide scarp line polygons were created from the initial scarp line inventory data provided by MDOT SHA. The scarp polygon identification instance segmentation machine learning model was trained using the inventory scarp line data. The trained scarp line detection machine learning model was subsequently applied to 1-m resolution statewide LiDAR DEM data to detect scarp polygons in Maryland. The research team also performed parametric study for the machine learning model to determine its optimal hyper-parameter values. To further fine-tune the machine learning model, the zonal statistic tools in acr gis pro were applied to calculate the terrain features for the 3-m wide scarp line polygons, and their distributions were visualized with plotted histograms.



Figure 24. Flowchart of machine learning-enabled automated scarp polylines generation

The MDOT SHA provided the initial scarp line inventory dataset which contains 1,438 polylines representing scarp lines distributed over the entire Maryland state. Sample scarp lines are shown in Figure 25 below. To use the instance segmentation machine learning model for scarp line polygon detection in Arc gis pro, the UMD team first converted the scarp polylines into scarp line polygons for machine learning training data generation. Scarp line polygons are created from scarp line (located as the center line of the polygon) with a given buffer width. Three different buffer width values (2m, 3m, 4m each side of scarp line as buffer width respectively) were considered (shown in Figure 25 below) and instance segmentation machine learning model for varying width scarp polygons from the four western counties (GA, WA, AL and FR counties) were trained based on 1-m resolution statewide LiDAR DEM data. The scarp line dataset

including 660 scarp lines were from the four western MD counties (GA, AL, WA, FR county). The first round machine learning model only includes scarp line polygons from the four western Maryland counties (GA, AL, WA, FR) and the detection is done using the slope raster tiff images.



Figure 25. Scarp line polygons with three different buffer width values (2m, 3m, 4m) for machine learning enabled scarp line detection

45-degree-increment rotation augmentation was also applied to expand the training data size from 1k to over 9k. Rotation augmentation would also help improving the machine learning model performance. All three machine learning models can output scarp line trial detection results during the training output. However, the 2-m buffer width machine learning model quit model training prematurely at Epoch no. 23. It is worth noting that machine learning model with 1-m buffer width was also trained but could not make any predictions during training and thus is removed from further consideration in this study. Both training and detection require over 10-hours computing time for a case with 9,000 training chips created from the 4 western Maryland counties' 1m-resolution slope raster training data. Based on the observed results of detected scarp line polygons for the four western Maryland counties (GA, AL, WA, FR) using the trained machine learning model, 3-m buffer width was determined to be the best and adopted for this study of scarp line detection using machine learning model. The computing time for machine learning scarp line detection for the 4 western counties was 12 hours 53 minutes. The machine learning model detected 115,176 scarp line polygon candidates for 4 western counties.

It is worth noting that the difference between the Arc gis slope tool and surface parameter slope tool was investigated by comparing their respective values at scarp line. For the slope scarp line polygon detection using machine learning model, training chips were first generated using the

slope raster tiff images as training data and the machine learning model’s hyper-parameters were tuned to optimize its performance for scarp line detection in Maryland. However, it was recognized that the slope function in arc gis pro substantially underestimates those large slope values typically associated with scarp lines (i.e., cliff-like abrupt drop in elevation at scarp line). The current slope raster used for training data generation is calculated from 1m-resolution LiDAR DEM by taking spatial derivative over a 3x3 cell area in arc gis pro, and this averaging operation would introduce large error to slope raster tiff at locations with abrupt change in elevation such as scarp line and result in low detection accuracy when used for instance segmentation machine learning model training for scarp polygon detection. As shown in Figure 27 below, the zonal statistics table for max slope for all scarp polygons in the four western Maryland counties (GA, AL, WA, FR counties) suggests the surface parameter slope tool generates larger slope values which is more consistent with actual slope values of scarp lines. This can also be explained using a sample slope shown in Figure 26. The slope tool in arc gis pro indicates the Slope value is 32 degree ($= \arctan(5/8)$), while the actual Maximum slope is 78.69 degree. This is because the slope tool in arc gis pro uses the average value of a 3x3 cell surface scanning window to calculate the slope value. For locations with abrupt change in elevation typical of scarp lines, its large slope at the scarp would be substantially underestimated due to this averaging action. Surface parameter slope raster and focal range statistics raster are thus adopted as the input feature for the re-trained machine learning model for detecting scarps.

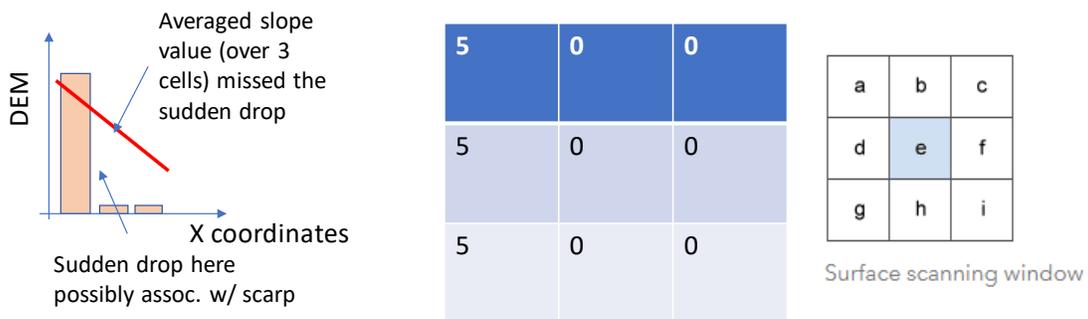


Figure 26. Illustration of the Slope tool in arc gis pro

The Surface parameter slope tool allows for flexible moving window size and adaptive window options that evaluates the local variability of the terrain and identifies the largest appropriate neighborhood size for each cell. Arc gis recommends using surface parameter slope tool if the input raster parameter value is high resolution with a cell size of less than a few meters. So in

this study the input value calculated using the surface parameter slope tool in arc gis pro is recommended as input feature data for machine learning model instead of slope raster images.

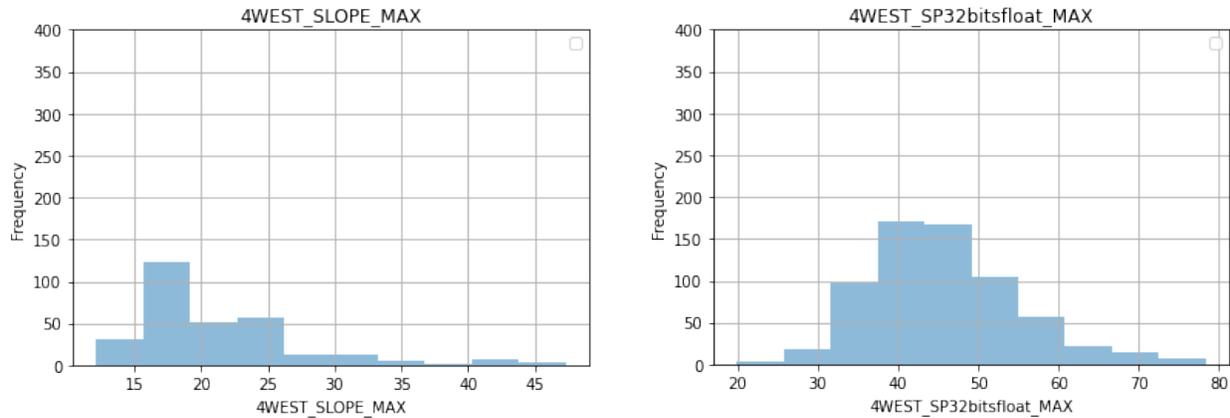


Figure 27. Zonal statistics (maximum slope) histogram of scarp polygon’s slope values: (a) Results from arc gis Slope tool; (b) Results from arc gis Surface Parameter slope tool

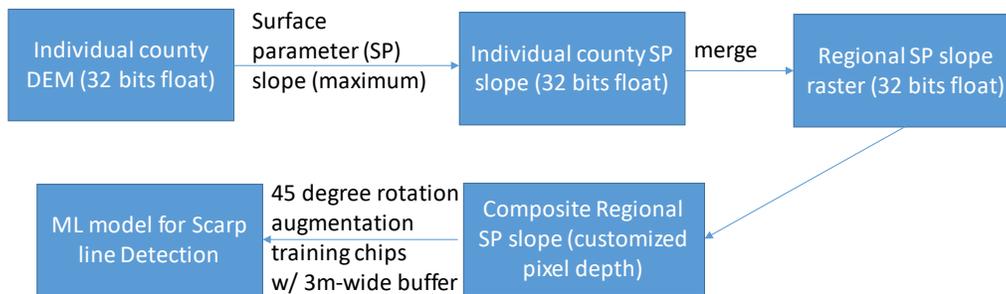


Figure 28. Flowchart of instance segmentation machine learning model based scarp line detection using unscaled 8-bits unsigned 1-m surface parameter slope raster tiff data

Next, new machine learning model has been trained using expanded data that merges the 1m-resolution LiDAR DEM files for the 11 counties in the middle region of Maryland. Surface parameter slope raster tiff images of this middle region were first generated from 1-m resolution LiDAR DEM data. The training chips generated from these surface parameter slope raster images used 3m-wide buffer scarp line polygons. The computing time for scarp line detection for the 11 middle counties took 25 hours 9 minutes. The instance segmentation machine learning model detected 90,028 scarp line polygon candidates in the 11 middle counties in Maryland.

Another bigger instance segmentation machine learning model for scarp line identification was trained in arc gis pro based on a consolidated dataset with a total of 9,336 training chips created

from the 4 western Maryland counties (GA, AL, WA, FR counties) and 1,0956 training chips created from the above-mentioned 11 middle region counties with 45-degree rotation augmentation. This machine learning model is called concatenated ML model. The total training time took approximately 10 hours 40 minutes for the instance segmentation model using the combined middle region and western region data. The trained machine learning model was then used to detect scarp line polygons in the west and mid region of Maryland by setting the confidence threshold equal to 0.3. The training loss curve for this 3m-wide scarp line polygon machine learning model is shown in Figure 29 below.

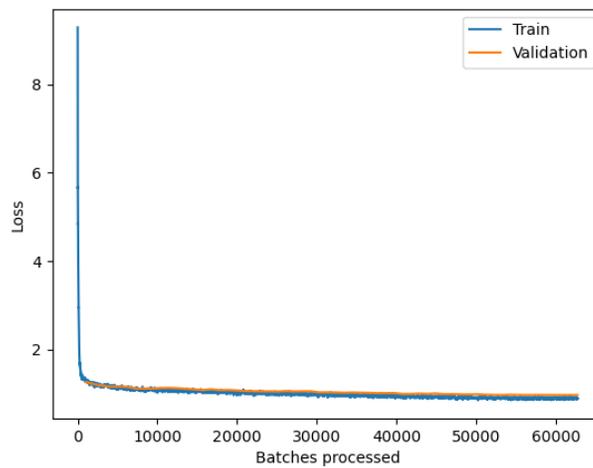


Figure 29. Training loss curve for 3m-wide scarp line polygon detection machine learning model

For those machine learning model detected scarp line polygons from the western and mid region surface parameter slope raster dataset, they were further processed by applying the boundary dissolve tools, creating centerline features for the polygons and filtering with 600-ft wide route buffer zone from roadway centerlines as shown in Figure 30. Quality check was also performed on scarp line candidates generated from ML model by visually checking the identified scarp lines and compared the quantities overlapped with inventory data and plotting histograms of total length of those identified scarp lines in the 4 western (GA, AL, WA, FR) and 4 middle (PG, BA, CH, AA) counties in Maryland based on two different confidence score threshold values.

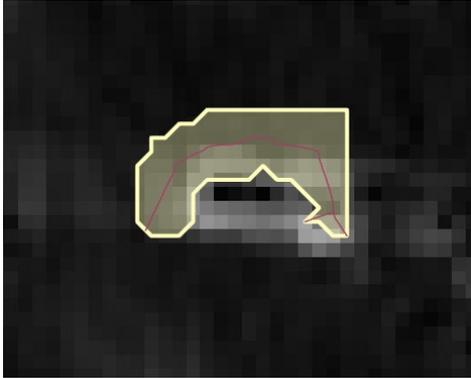


Figure 30. Sample detected scarp line by the instance segmentation machine learning model (yellow polygon = detected scarp polygon, scarp line is the center line of the yellow polygon)

A parametric study was performed to see if the model performance can be further improved by increasing the batch size to 16 (currently 8) and test the training strategy of extending the total training epochs to 100. Additionally, a separate machine learning model was trained with only middle region data, but no significant improvement was observed. Figure 31 shows the training loss curve of the scarp line detection machine learning model for the middle region counties. A parametric study of pixel depth was performed by comparing the detection performance of 32-bits float, 8-bits unsigned and 16-bits unsigned input feature raster tiff images as training chips. 16-bits unsigned surface parameter slope raster were generated for the re-trained machine learning model for scarp detection.

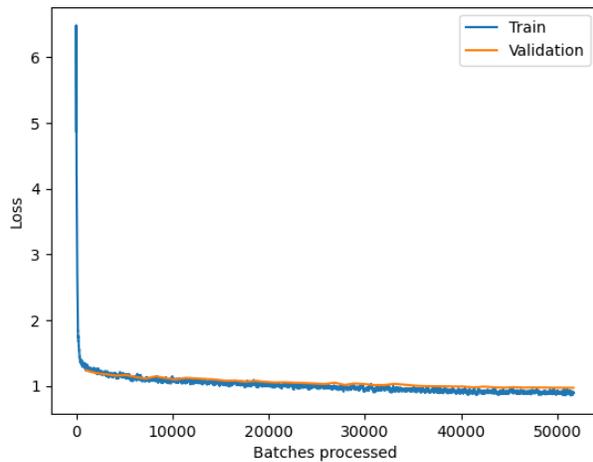


Figure 31. Training loss curve of scarp line detection machine learning model for the middle region counties

Post-processing of the detected candidate scarp polygons is required as several small polygons might partially overlap to form the true scarp polygon. Therefore, the following post-processing steps are recommended: 1. Filtering the detected candidate scarp polygons with a pre-specified

confidence score threshold (e.g., 0.6) and filtering by 600-ft buffer zone on both sides of highway center lines. In general, using higher confidence score threshold results in less number of identified scarp polygons. 2. Applying the dissolve tool to merge the partially overlapped smaller polygons at the true scarp polygon, or separating them by applying multipart to single part tool. 3. Applying the polygon to center tool to generate the scarp line.

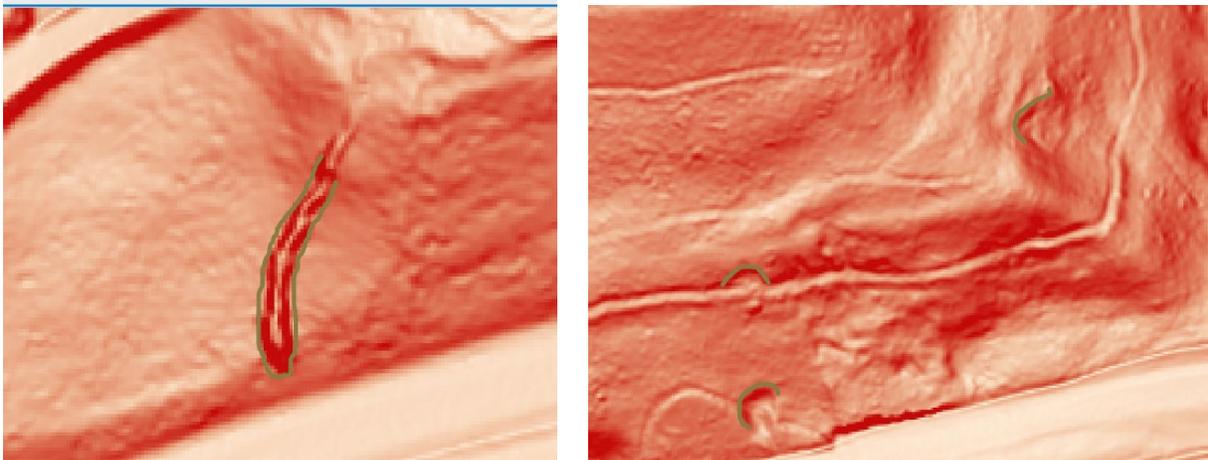


Figure 32. Machine learning detected scarp line samples in 4 western counties of Maryland

To show the machine learning scarp line detection results, the concatenated machine learning model was implemented to identify the scarp lines in the 4 western counties (GA, WA, AL, FR). The total computing time for running this machine learning model took 15 hours 5 minutes for these four counties. A total of 24,770 scarp centerlines were generated from detected polygons with confidence score threshold set to be 0.6. However, after applying the filtering step with 600-ft buffer zone on both sides of highway center lines mentioned above, only 2,831 scarp centerlines were kept inside the 600-ft route buffer zone: 537 scarp lines were in the GA county, 436 scarp lines were in the WA county, 1242 scarp lines were from AL county, while the remaining 616 scarp lines were from FR county. Table 5 lists the quantity of machine learning detected scarp lines in 4 middle region counties (AA, BA, PG, CH) with confidence score threshold set to be 0.4. It can be seen that machine learning models can detect additional scarp lines not available in the initial SHA inventory dataset. It should be noted that this initial SHA inventory dataset was generated only for machine learning training purpose as seed dataset and was not intended to be exhaustive and complete. Therefore, it is normal that additional new scarp

lines are generated by running the machine learning model. Figure 32 shows sample scarp lines detected in these 4 western counties of Maryland.

Table 5. Quantity of machine learning detected scarp lines in 4 middle region counties (AA, BA, PG, CH) (Confidence score threshold = 0.4)

County	# of ML detected scarp lines inside 600-ft route buffer zone from highway center line	# of ML detected scarp lines intersecting with MDOT SHA inventory scarp lines
AA	2,029	34
PG	1,545	20
CH	5,18	2
BA	1,960	40

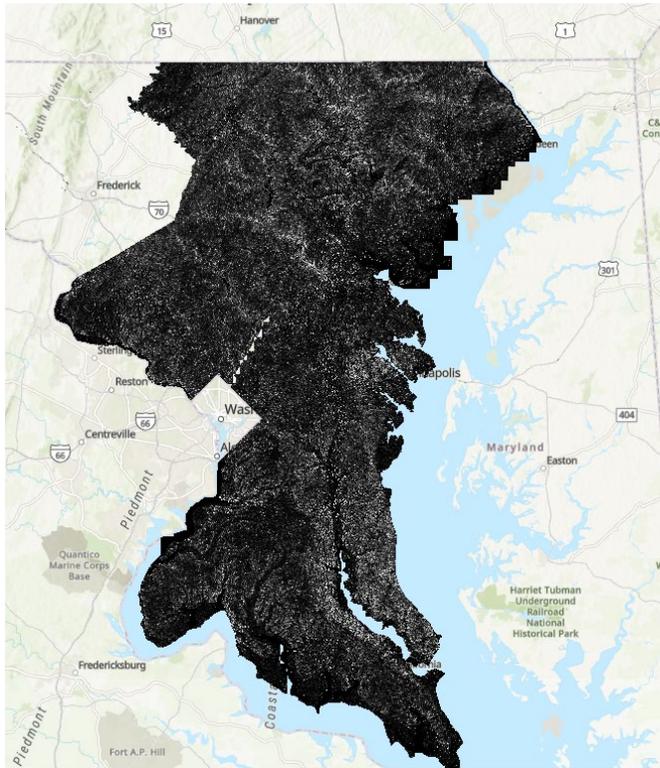


Figure 33. Maps of training chips from 11 middle region counties generated from 16-bits unsigned surface parameter slope raster tiff images

Machine learning model for scarp line detection was subsequently run on the 16-bits 3-band surface slope parameter raster tiff data for another nine counties (MO, HO, HA, CL, CA, CO,

CH, FR and SM counties), followed by applying the above-described post-processing steps on the detected scarp polygons. Detections are done for 3-band 16 bits unsigned surface parameter slope rasters of MO, HO, HA, CL, CA, CO, CH, FR and SM counties. The machine learning detected scarp line results were compared with initial scarp line inventory data, as shown in Figure 35. The average running time for each county took approximately 3.5 hours when confidence score threshold is set to be 0.3. Sample scarp line detection results (in raster image format) of SM counties are also shown in Figure 34. Table 6 below summarizes the total number of scarp lines detected by machine learning model for these 9 counties. The detection is based on a confidence threshold of 0.3 and 1200-ft buffer zone from the route centerline. Samples of detected scarp lines after post-processing are shown in Figure 35 with the slope rasters as background in arc gis pro.

Table 6. Quantity of machine learning detected scarp lines for nine counties (MO, HO, HA, CL, CA, CO, CH, FR, SM counties) (Confidence Score threshold = 0.3)

County	# of detected scarp lines within 1200-ft buffer zone of highway centerline
MO	2574
HA	1185
HO	2047
CL	1085
KE	486
CA	486
CE	1878
CH	1178
SM	1256

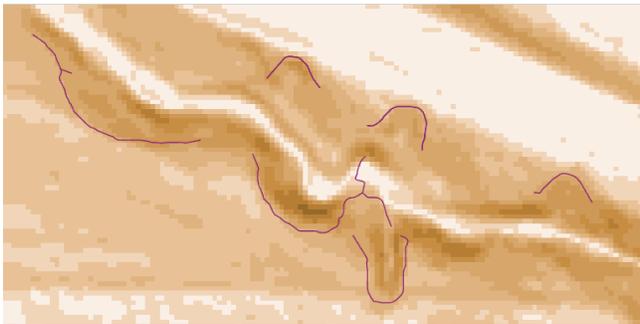
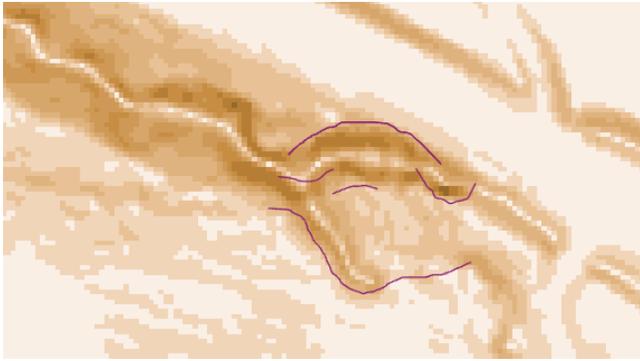


Figure 34. Samples of machine learning detected scarp lines in SM county

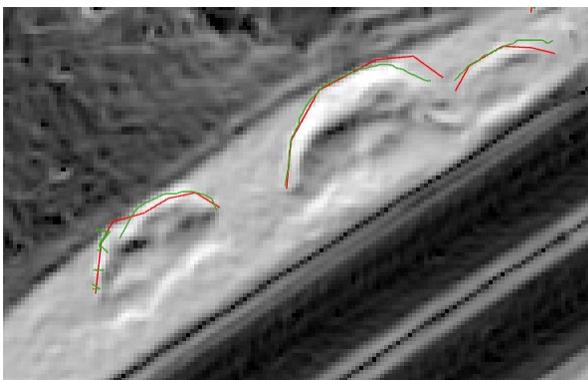
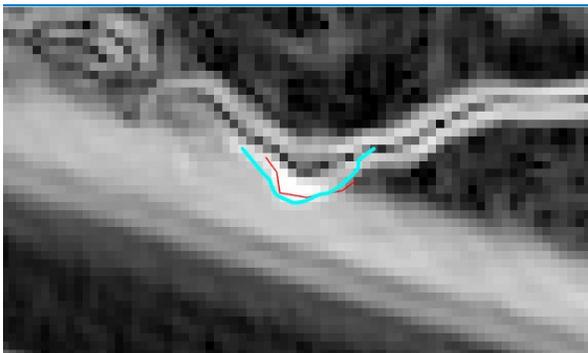


Figure 35. Samples of machine learning detected scarp lines (red lines = inventory scarp line data; blue or green line = machine learning detected scarp lines)

CHAPTER 5. DEVELOPING NEURAL NETWORK MACHINE LEARNING MODEL FOR CONCRETE STRENGTH TEST DATASETS

The research team trained fast.ai neural network model for concrete compressive strength tabular datasets and used the trained models to predict concrete strength with given feature variable values. Tuning the neural network model for optimal hyper-parameter values was conducted. Comparison of the predicted results from Fast.ai neural network model and real test data was also made by calculating the corresponding confusion matrices and drawing scatter plot in this study. The research team also did literature review to ensure the tabular data neural network model is current and has been updated for the concrete compressive data based on parametric study of hyper-parameter values.

Concrete compressive strength predictions were made by machine learning models developed using thousands of historic strength tests with two different mix designs and several concrete placement variables. Machine learning models were trained for both Mix#3 dataset and Mix#6 dataset. Data cleansing was conducted to ensure that only the most reliable data was used in training the tabular machine learning regression model. The first step of data preprocessing is to break down concrete mix design code into multiple feature sections including region code (RC), mix strength code (MSC), admixture code (AC), component code (COMC), regional plant number (RPC) and consecutive mix code (CMC). Records with missing feature values or with non-standard format were discarded from the dataset. Feature variables used to predict the concrete strength include the water-cement ratio (WCR), entrained-air (ENTAIR), ambient temperature (ATEMP), concrete temperature (CTEMP) and all features generated from the mix design code. The target features are the 28-day compression strength of standard 6"-diameter concrete cylinder specimens. Preliminary machine learning models were first built to predict the compressive strength of concrete cylinder #1 and average strength value of cylinders #1 and #2.

The training and validation dataset only includes data records with feature variables in the reasonable range. The reasonable ranges of concrete mix and casting feature values for the two concrete mix designs (Mix#3 and Mix#6) considered in this study are listed in Table 1. Restrictions on the lower and upper bound values of considered feature variables were

determined based on MDOT SHA staff's advice and the actual statistical distribution of training dataset. For Mix#3 concrete data, a total of 17,612 records were available for machine learning model training after data preprocessing and cleansing steps. For example, as shown in Figures 36 and 38, most of the test data are within the range of 3,950 to 5,390 psi before pre-processing. However, the maximum strength is 53,120 psi which is unrealistically high and the minimum strength is 0 psi (too low value). After data cleansing and filtering, the real strength distribution for real Mix#3 concrete data ranges from 1260 to 5,990 psi (it is noted that no data samples between 5991 psi to 7500 psi). Similarly, most of strength value for Mix#6 dataset lies in the buckets from 4670 to 5,750. After data preprocessing and cleansing, the strength value range for Mix#6 dataset is from 3,050 to 10,000 psi.

Table 7. Feature value range of concrete strength data for machine learning model training

Features	Value Range of Feature Variables
WCR	0.3-0.6
SLUMP (in)	0-9
ENTAIR	0 - 0.15
ATEMP (°F)	18 -105
CTEMP (°F)	50 -97
Strength of CYL1, CYL2 (psi), Mix#3	1,000 – 7,500
Strength of CYL1, CYL2 (psi), Mix#6	3,000 – 10,000

The architecture of the adopted feedforward neural network model is shown in Figure 42. A typical hidden layer with ReLU (rectified linear unit) activation function, batch normalization, and dropout is the state of art for feedforward neural network model. Batch normalization proposed by Ioffe and Szegedy (2015) normalizes the output of a previous activation layer by subtracting the batch mean and dividing by the batch standard deviation. This ensures that the gradients are more predictive and thus allows for use of larger range of learning rates and faster network convergence (Santurkar et al. 2018). Therefore, Fast.ai-adopted neural network model is already the state of art model for tabular data.

Two preliminary tabular models were developed for mix 6 dataset. The target variable is the compressive strength of cylinder 1 concrete for model 1 and the average strength value of cylinder 1 and cylinder 2 for Model 2. Features used as input variables are 'RC', 'AC', 'COMC', 'MSC', 'RPC', 'CMC', 'SLUMP', 'WCR', 'ENTAIR', 'ATEMP', 'CTEMP', and 'DPLACED_MONTH'. Two hidden layers with 200 nodes in each layer were used to connect the input layer and output layer. The training dataset has 8,692 samples, with 10% allocated for the validation set. Batch size of 128 was used for both models. The optimal training epoch number of 180 was used for both machine learning models to avoid overfitting. A prediction test was done on the Mix#6 dataset. A scatter plot is used to show the relationship between machine learning predicted data and real data as shown in Figure 45. The model has an RMSE (root-mean-square error) of 511.92 and R^2 score of 0.72 between predicted data and real data for Model 1. The machine learning models which predict the compressive strength of Mix#3 concrete data used the same input feature variables and neural network hyper-parameters, with epoch number set as 120. The prediction test shows an RMSE of 602.8 and R^2 score of 0.55 between the predicted data and real data for this preliminary machine learning model using Mix#3 data. These machine learning models need to be improved so both machine learning models can have a R^2 score over 0.8. Strategies for improving the machine learning model were evaluated from literature review study and selected strategy appropriate for concrete data use is discussed next.

The concrete dataset was observed to have data imbalance issues in compressive strength values, therefore the data resample strategy was applied to overcome this imbalance problem by improving the distribution and thus the model performance. Regression resampling is a technique used in machine learning to address the issue of imbalanced data where the distribution of the target variable is not uniform across different values or ranges. In regression oversampling, the minority class is oversampled to balance the distribution of the target variable (Luís Torgo et al. 2013). In this study, the samples are duplicated and replicated in a controlled manner to change the distribution of target values. For example, in Mix#6 dataset, data in the lowest bucket and highest bucket has been duplicated by five times while no replication was done in those middle range buckets. Other buckets have also been duplicated to change the value distribution as shown in Figure 50. A similar oversample strategy was also applied to the Mix#3 dataset as well. All data samples are divided in to 20 buckets. The lowest value bucket was

duplicated by 8 times and the middle peak value bucket was duplicated by 2 times. The value distribution after data resampling was shown in Figure 52.

Machine learning models were trained for the resampled dataset for Mix#3 and Mix#6. The resampled Mix#3 dataset includes a total of 49,872 samples, while Mix#6 has a total of 11,682 samples. The machine learning model was trained with 600 epochs to make sure training converged without overfitting or under fitting. Individual prediction test was made on the original dataset to predict the average strength of cylinder 1 and cylinder 2 specimens. Improved prediction performance was observed in both Mix#3 model and Mix#6 model as shown in Figure 55 - 56. The updated Mix#6 model has a RMSE of 332.64 and a R^2 value of 0.87, while the updated Mix#3 model has a RMSE of 230.4 and a R^2 value of 0.89. It is concluded both machine learning models for predicting the concrete strength significantly improved. Considering the impressively high R^2 value near 0.9 for both mix designs, these machine learning models are adopted for final use.

Next, to analyze the relative importance of each input features to the machine learning model prediction, SHAP (Shapley additive explanations) analysis was performed. It is a technique for explaining the output of a machine learning model by quantifying the contribution of each feature to the predicted outcome. The SHAP values decomposed the predictions into the of each input feature, which allows for understanding and interpreting the importance of each feature in the prediction, and to identify the most influential features that may affect significantly the machine learning predictions. The SHAP values are calculated by comparing the model's predictions with and without a particular feature. The difference in prediction is attributed to the feature's impact on the model's output. This process is repeated for each concrete-test related feature, and the resulting SHAP values can be visualized using in the summary plot.

Figures 57 to 58 show feature importance ranking using the SHAP value calculated for the data samples from the original Mix#3 and Mix#6 datasets, as well as for the original machine learning model and data-resampled model. Each horizontal axis represents an input feature variable. The variation on each horizontal scatter plot represents the variation to average value of concrete strength predictions to which the individual input feature contributes. The color of the scatter

points indicates the value of the feature for that particular instance, with red color indicating high values and blue indicating low values. The position of the scatter points indicates the direction of the feature's effect on the prediction, with features that increase the prediction shown to the right of the plot and features that decrease the predicted value shown to the left. For example, Figure 57 indicates higher ambient temperature reduced the 28-days compressive strength for the Mix#3 concrete, which is consistent with the findings by other researchers by Kaleta-Jurowska (2020).

The SHAP plot indicates the three most influential feature variables to the machine learning predicted concrete strength are ranked as follows: R_C, ATEMP, RP_C. COM_CRP_C, ENTAIR and ATEMP are the three most important influencing features for Mix#6 models. However, the plot also shows the importance ranking of feature 'R_P' for data-resampled model increased from 11 to 1 compared to the original model.

For the classification machine learning model to predict the confidence score of passing the 28-day concrete strength test, graphs of true value of concrete strength from laboratory test vs. confidence score of machine learning classification. The purpose of this classification machine learning model is that by looking at confidence score (e.g., below a certain threshold value), it is decided whether there are merits to do the actual strength test if the confidence score from machine learning model prediction is fairly low. A binary classification machine learning model was thus trained to output the confidence score of concrete specimens passing the 28-days strength test. For Mix#3 dataset, samples larger than 3000 psi were assigned the label "Pass" while samples smaller than 3000 psi were assigned "Fail". Data resampling strategy was also applied to Mix#3 dataset to overcome the data imbalance issue. The machine learning model used batch size of 2,048 and was trained for 200 epochs. The f1-score for both categories reached over 0.8 as shown in the accuracy report. For Mix#6 dataset, data was cut and assigned the pass/fail label by the strength threshold of 4500 psi. The f1-scores for both categories were close to 1.

The plot of true concrete strength vs. predicted passing confidence score was shown in Figures 63 to 64. It can be clearly seen in these scatter plot graph of true concrete strength value vs. machine learning prediction confidence score that most of the data are located either in Quadrant

#1 or Quadrant #3 of the graph while very small percentage of data points fall within Quadrant #2 for both Mix3 and Mix6 data. This distribution pattern suggests that if 0.5 confidence score is used as the dividing line between passing or not passing the strength test, those predictions with confidence score < 0.5 have a very high chance not to pass the strength. The ideal case is that all points are either in Quadrant #1 (pass) or Quadrant #3(fail the test). Very small percentage of points are in Quadrant #2 indicates the machine learning classification is doing its job in reliable prediction of concrete strength in most cases. No data points in Quadrant #4 suggests that predictions with confidence score > 0.5 always pass the strength test. Those points with strength > 7000 psi have a confidence score near 1.0 (i.e., almost certain to pass the test). Most of the data in Quadrant #3 has a low confidence score value (i.e., low confidence score suggests it is highly likely not to pass the test). The prediction achieved high accuracy with 99.51 for Mix#6 and 99.45% for Mix#3. For each mix strength, samples with a real concrete strength greater than the strength threshold (orange line in the figure) but with a label of failure (Quadrant #2) is less than 2% of total data samples in Quadrants 2 and 3 combined, which is useful for engineers to decide whether to throw out those specimens without tests when performing the compressive strength test.

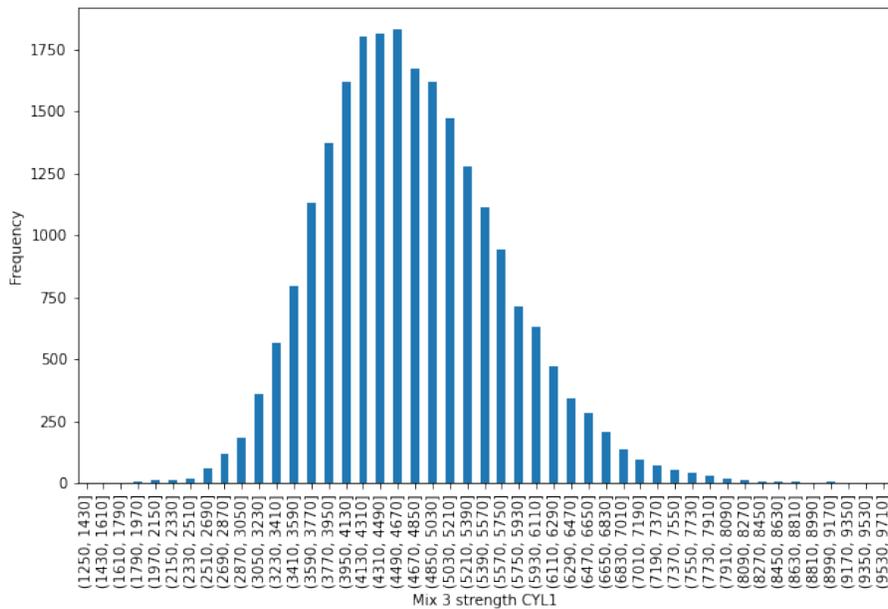


Figure 36 Histogram of Mix 3 Cyll original data

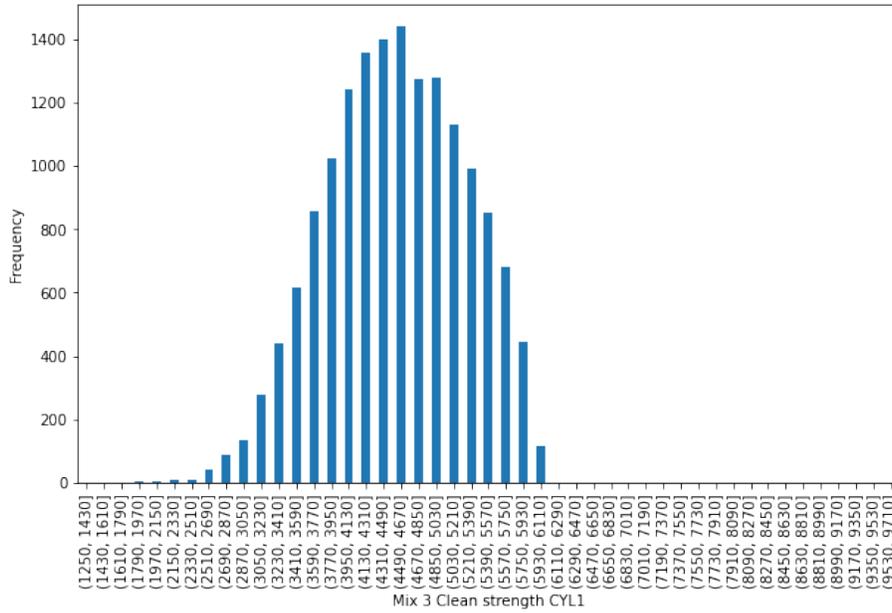


Figure 37. Histogram of Mix 3 Cyl1 clean data

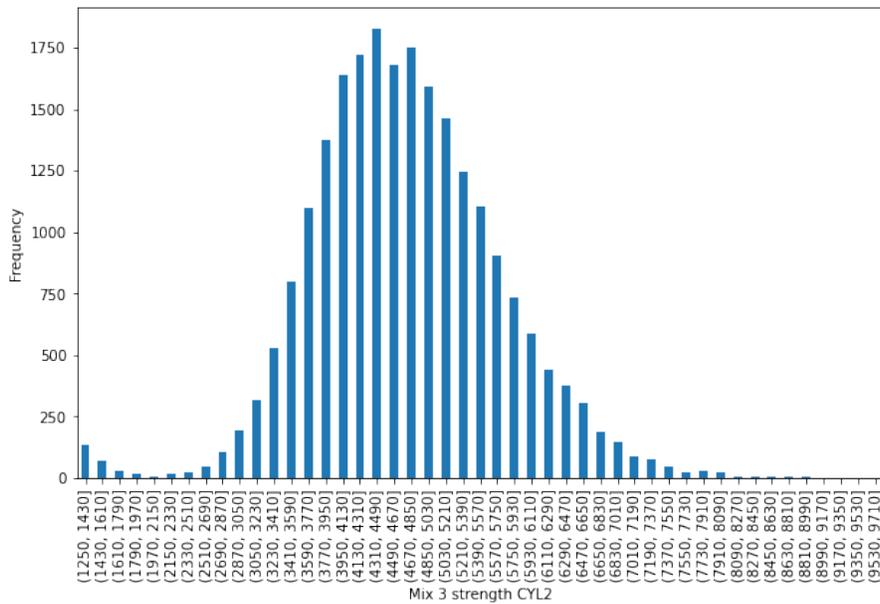


Figure 38 Histogram of Mix 3 Cyl2 Original data

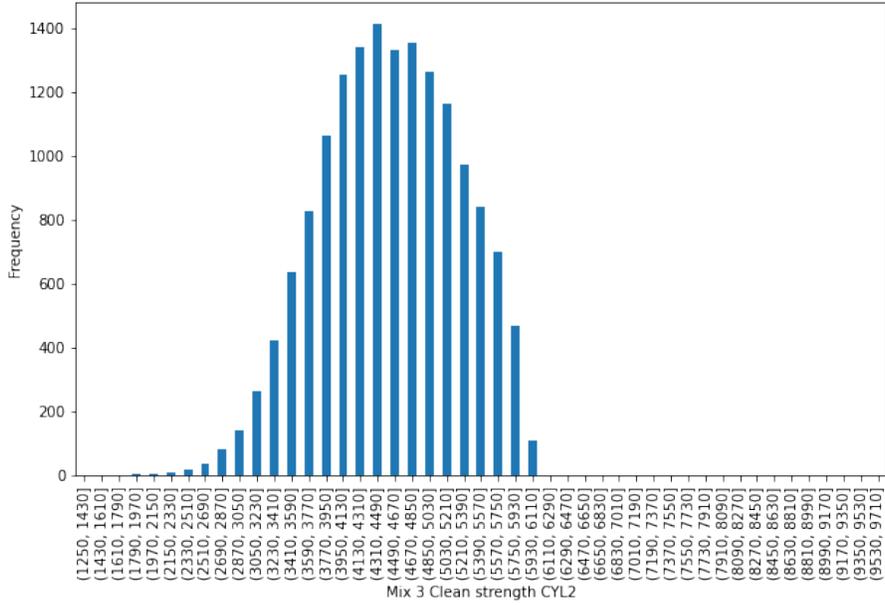


Figure 39 Histogram of Mix 3 Cyl2 clean data

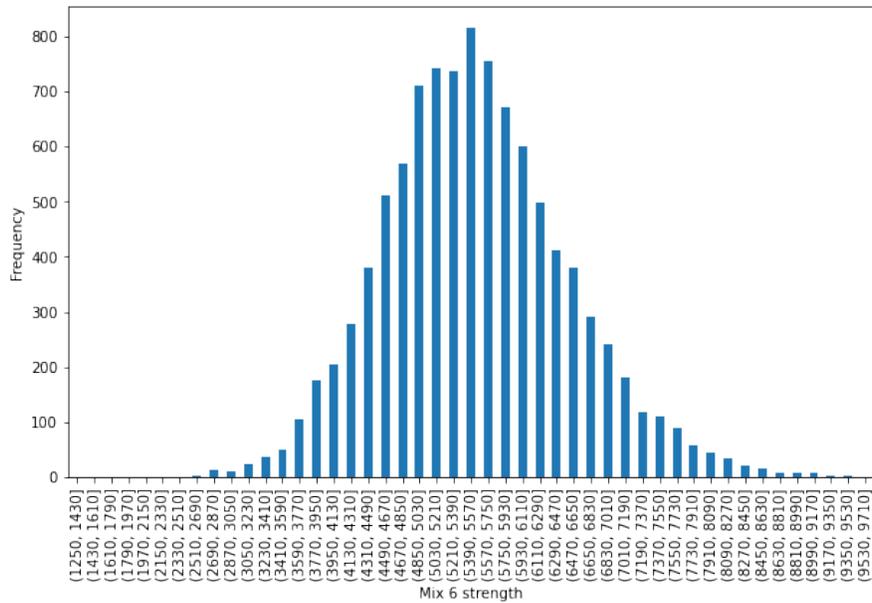


Figure 40 Histogram of Mix 6 Cyl1 original data

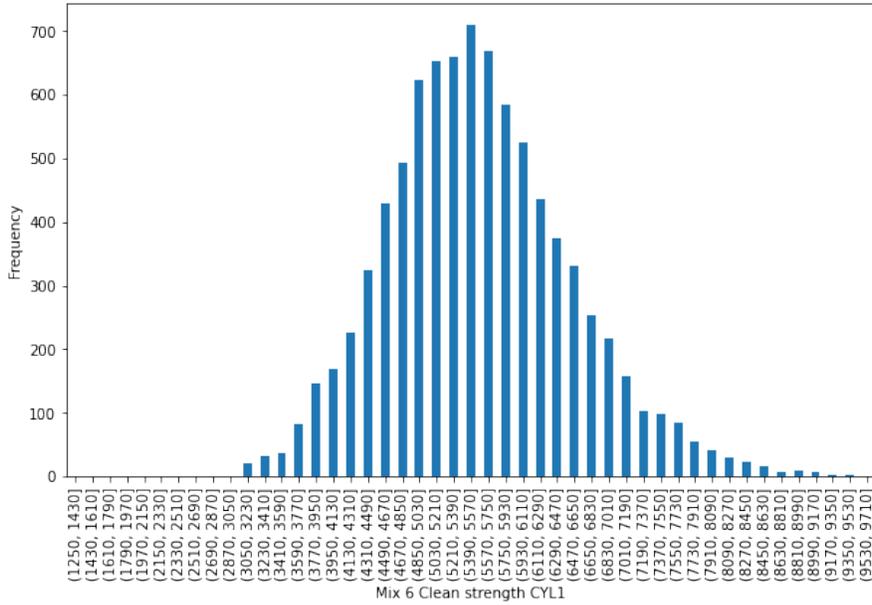


Figure 41 Histogram of Mix 6 Cyl1 Clean data

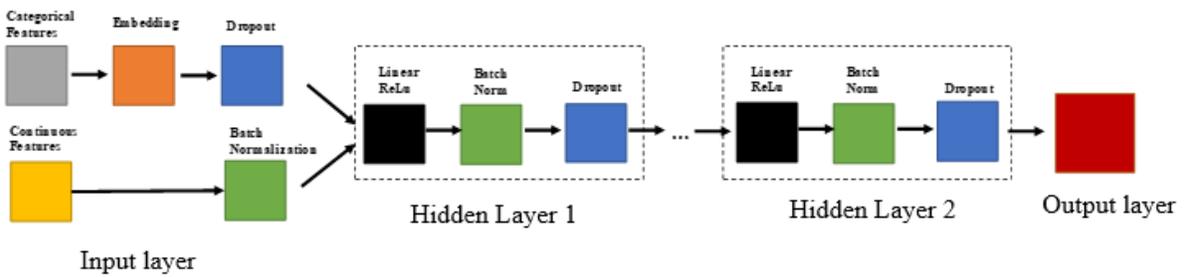


Figure 42 Architecture of feedforward neural network model for concrete strength data

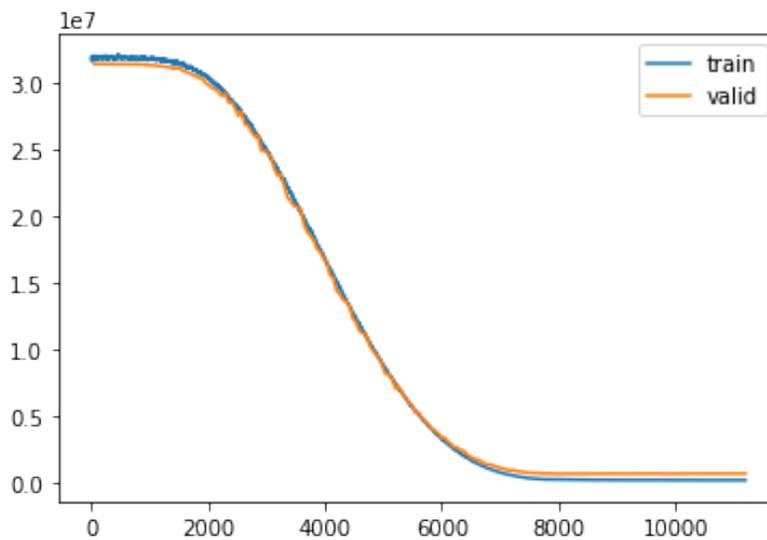


Figure 43 Training loss curve, mix 6, model 1

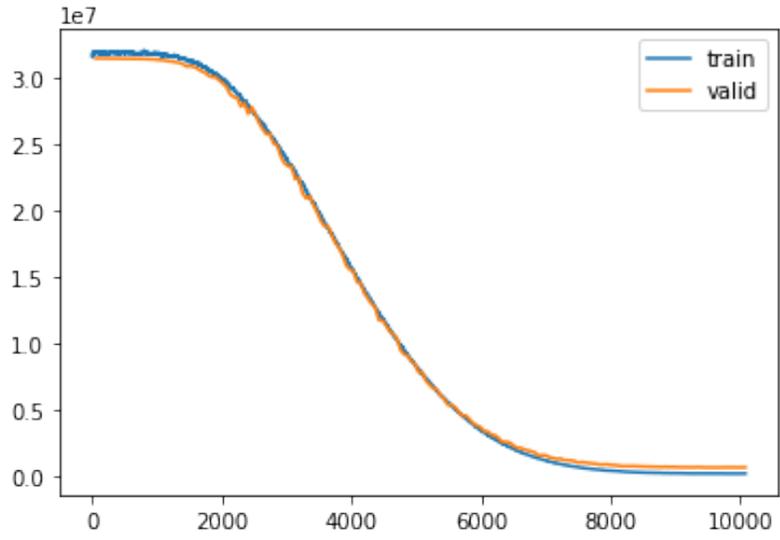


Figure 44 Training loss curve, mix 6, model 2

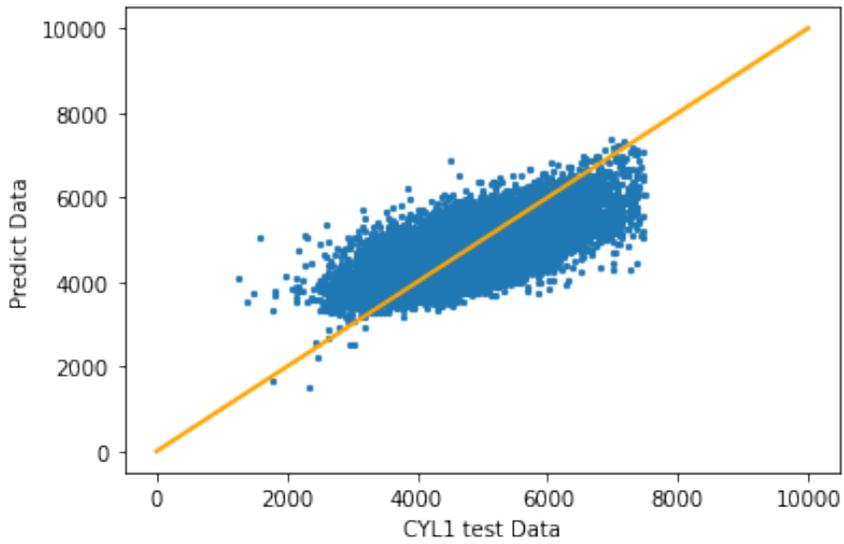


Figure 45 x-y scatter plot mix 3, model 1

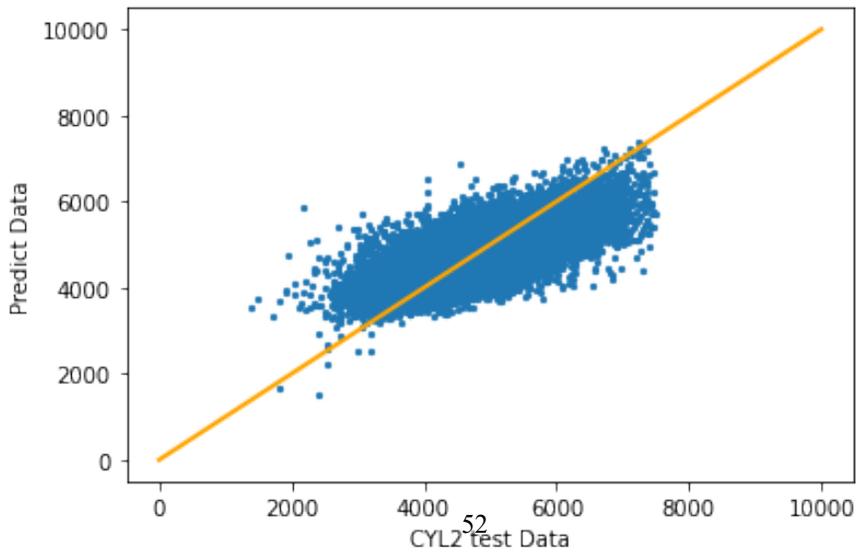


Figure 46 x-y scatter plot mix 3, model 2

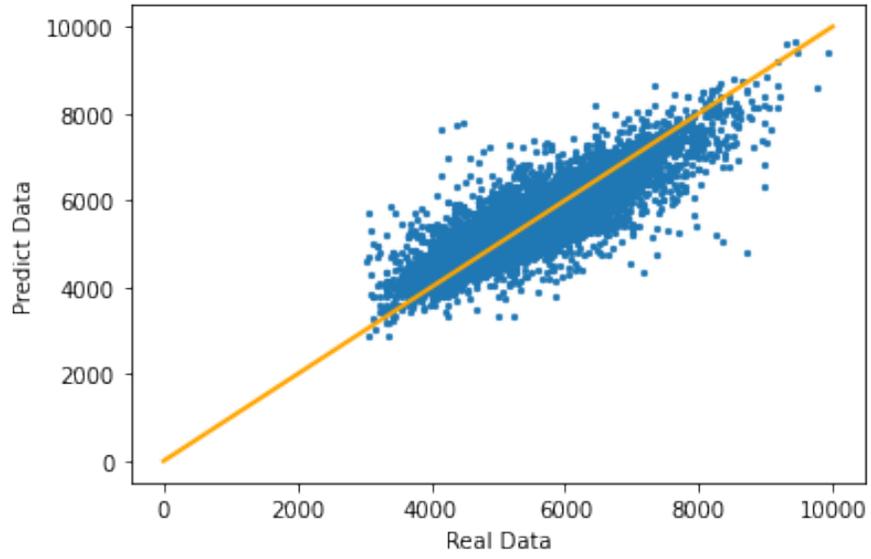


Figure 47 x-y scatter plot mix 6, model 1

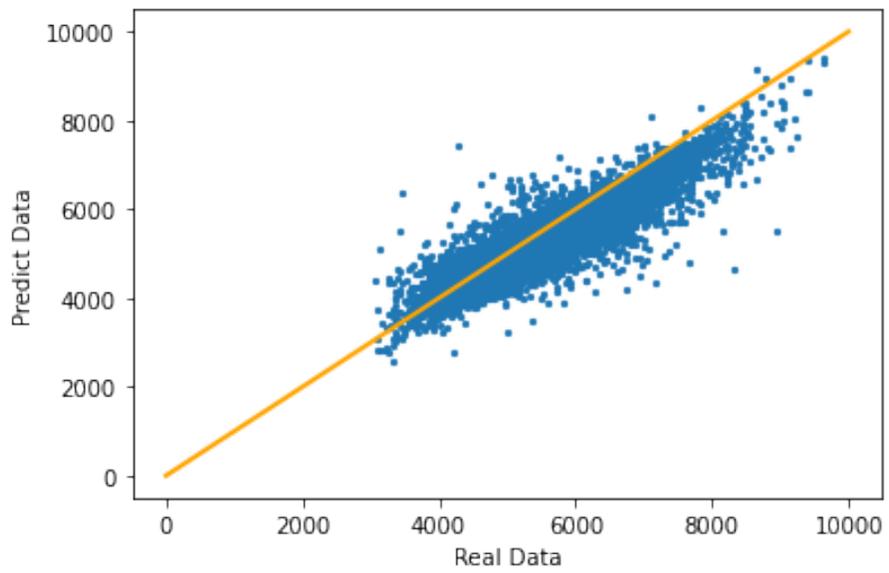


Figure 48 x-y scatter plot mix 6, model 2

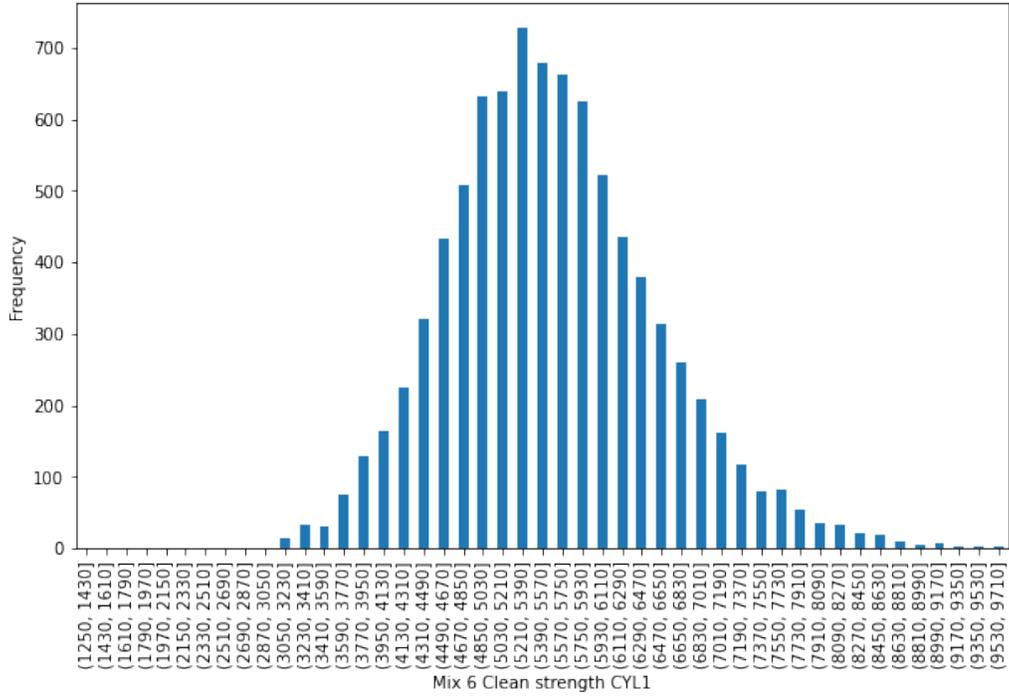


Figure 49 Original distribution of Mix 6 Clean strength Mean CYL1 CYL2

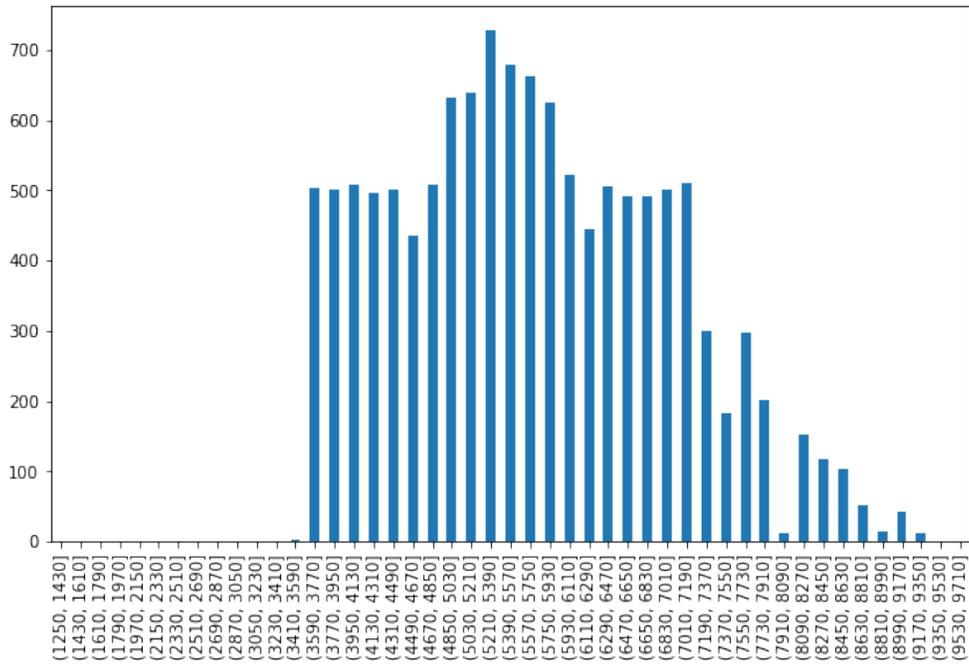


Figure 50 Resampled distribution of Mix 6 strength Mean CYL1 CYL2

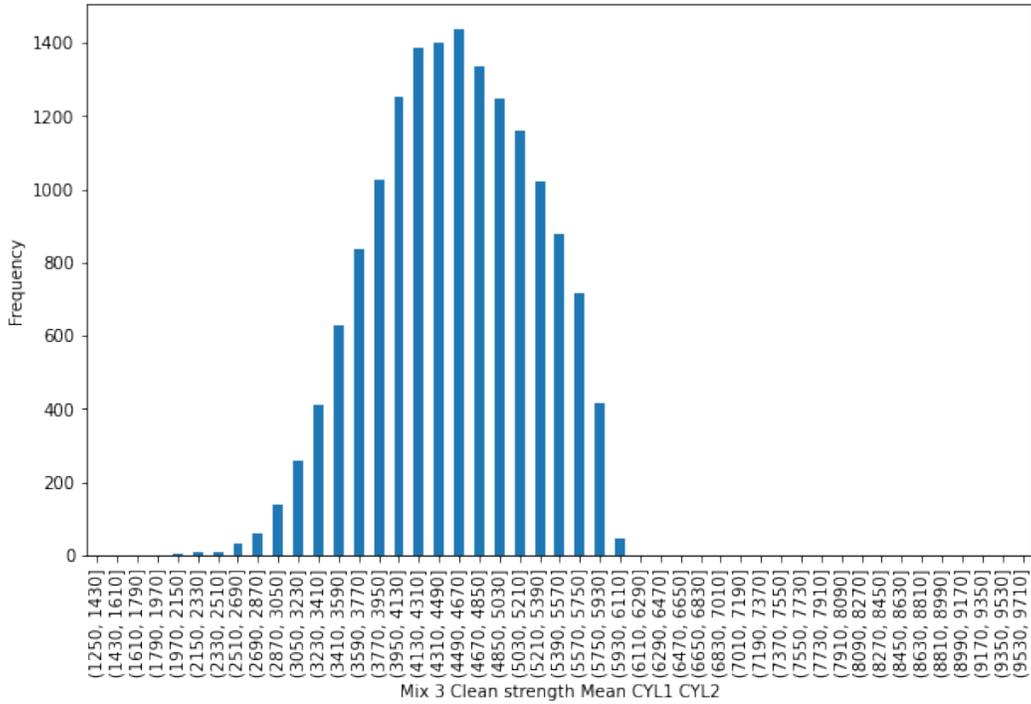


Figure 51 Original distribution of Mix 3 Clean strength Mean CYL1 CYL2

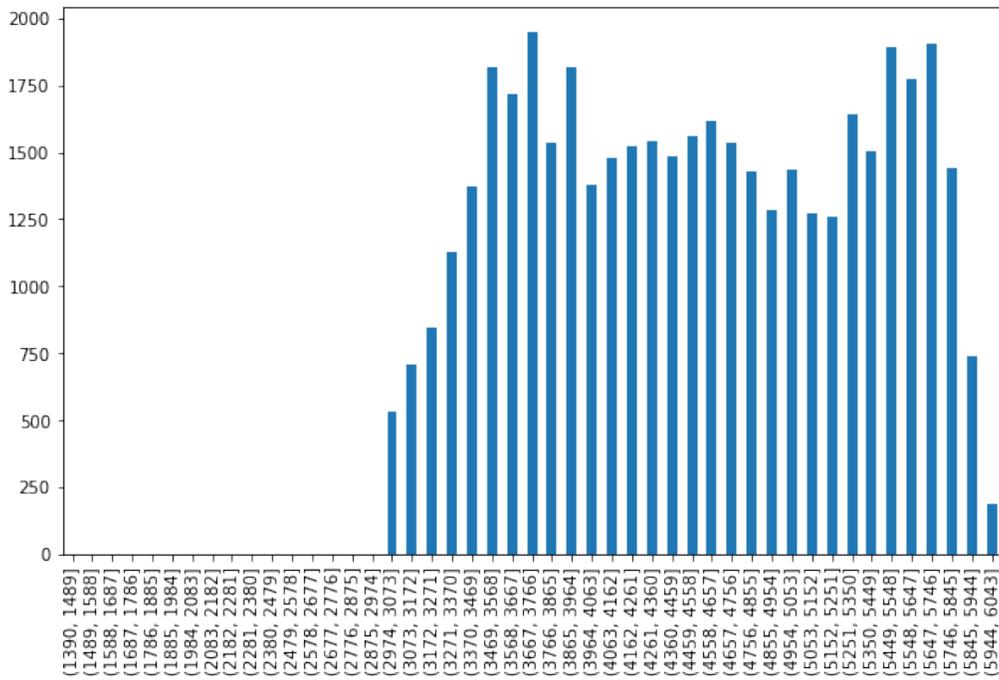


Figure 52 Resampled Mix 3 strength Mean CYL1 CYL2

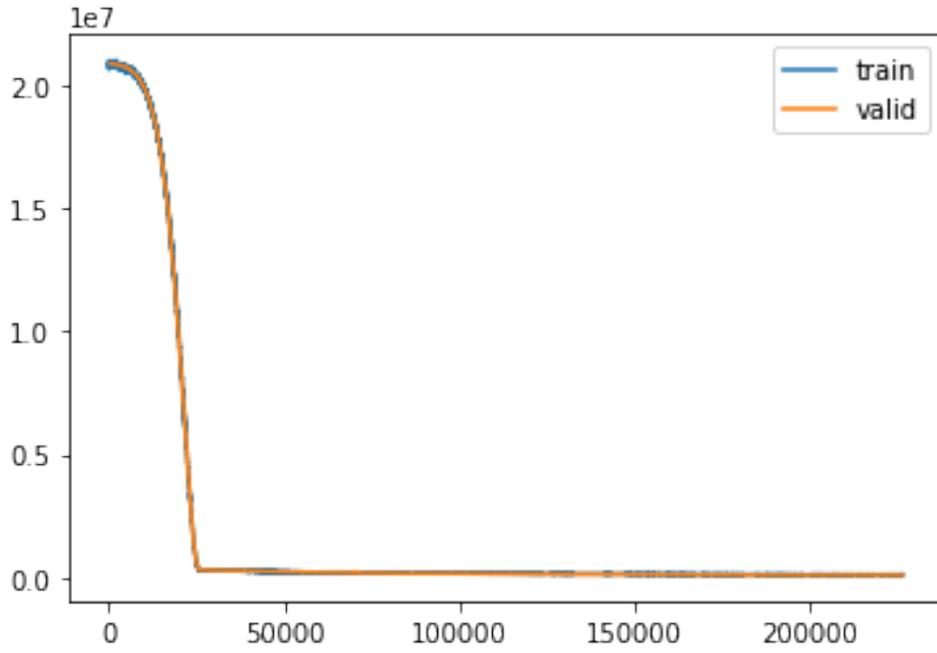


Figure 53 Mix 3 model training loss curve

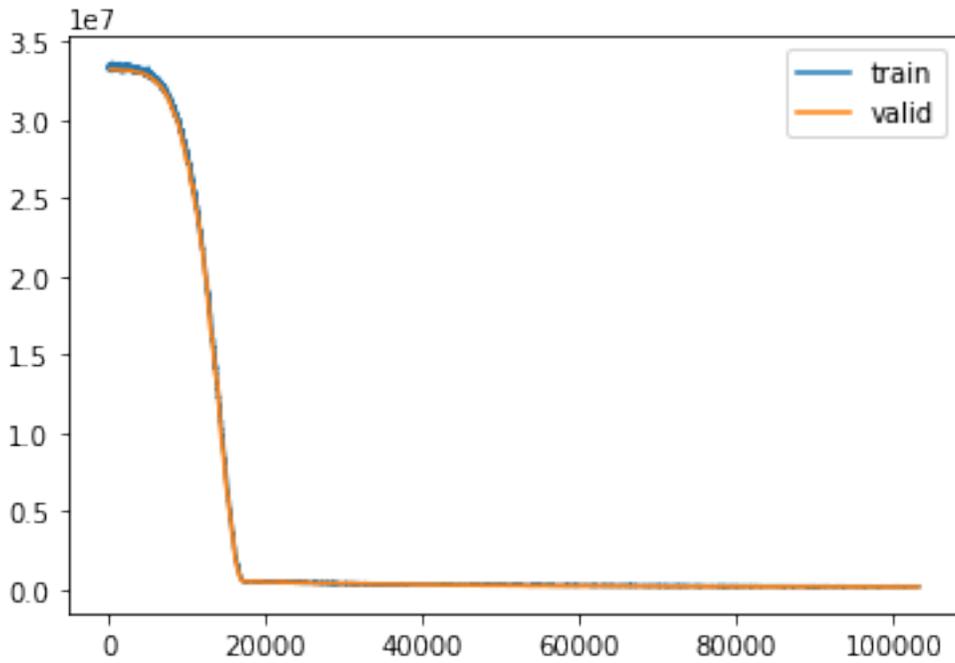


Figure 54 Mix 6 model training loss curve

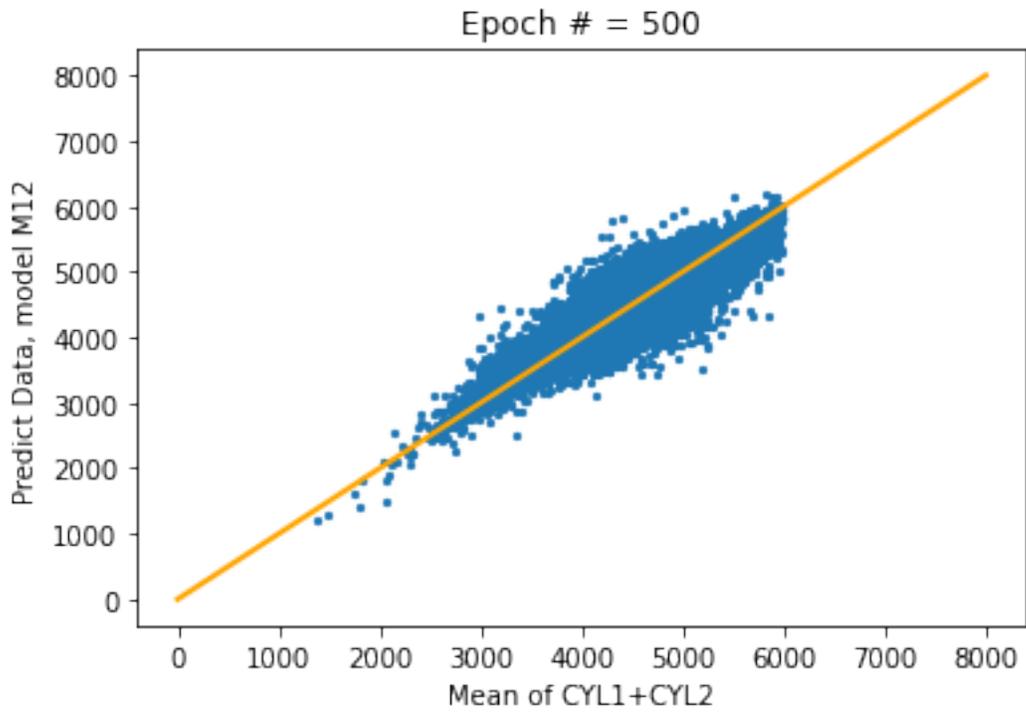


Figure 55 x-y scatter plot for mix 6 mean cyl1 cyl2

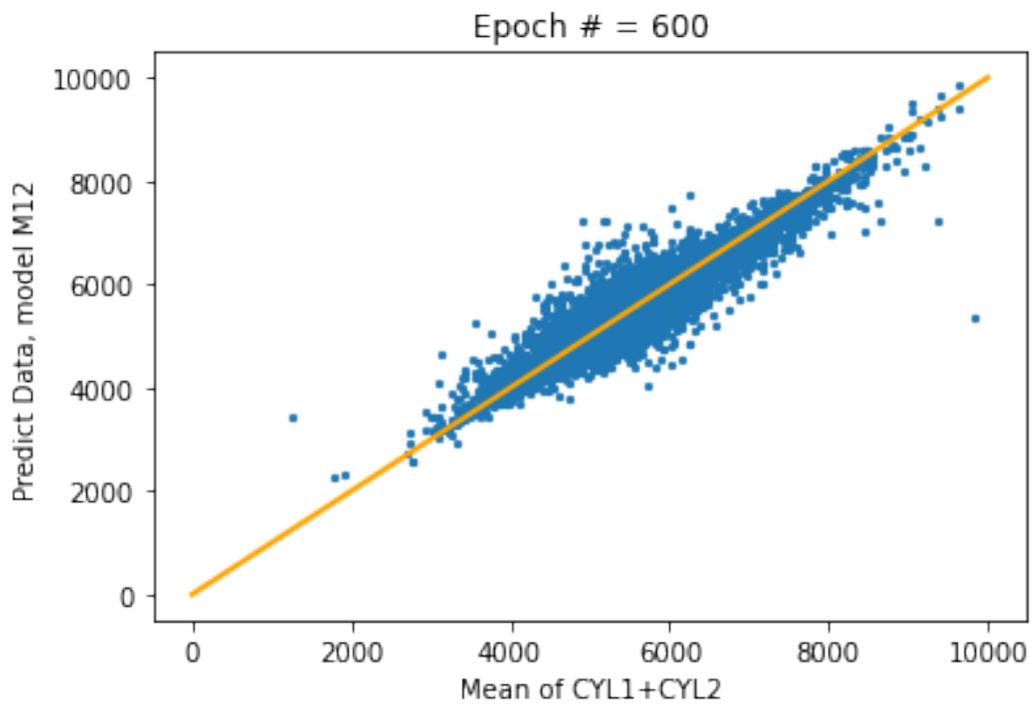


Figure 56 x-y scatter plot for mix 3 mean cyl1 cyl2

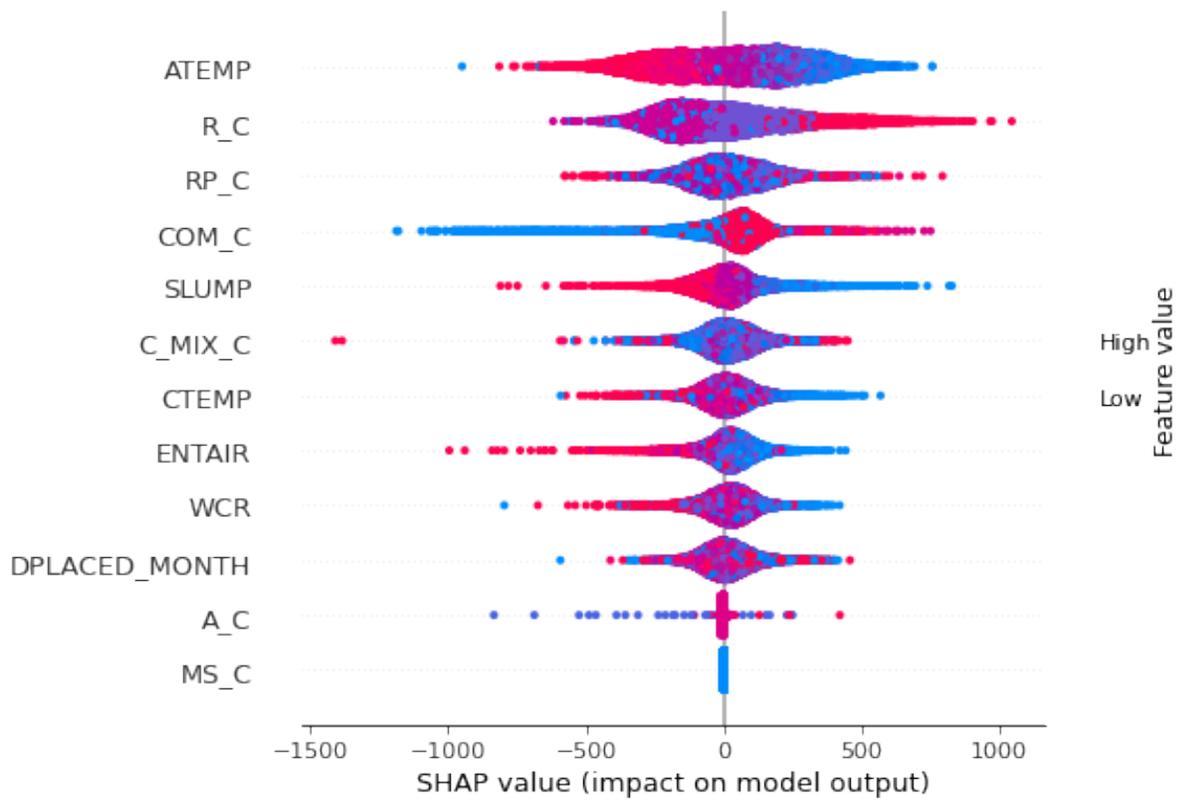


Figure 57 SHAP plot for resampled dataset, mix 3

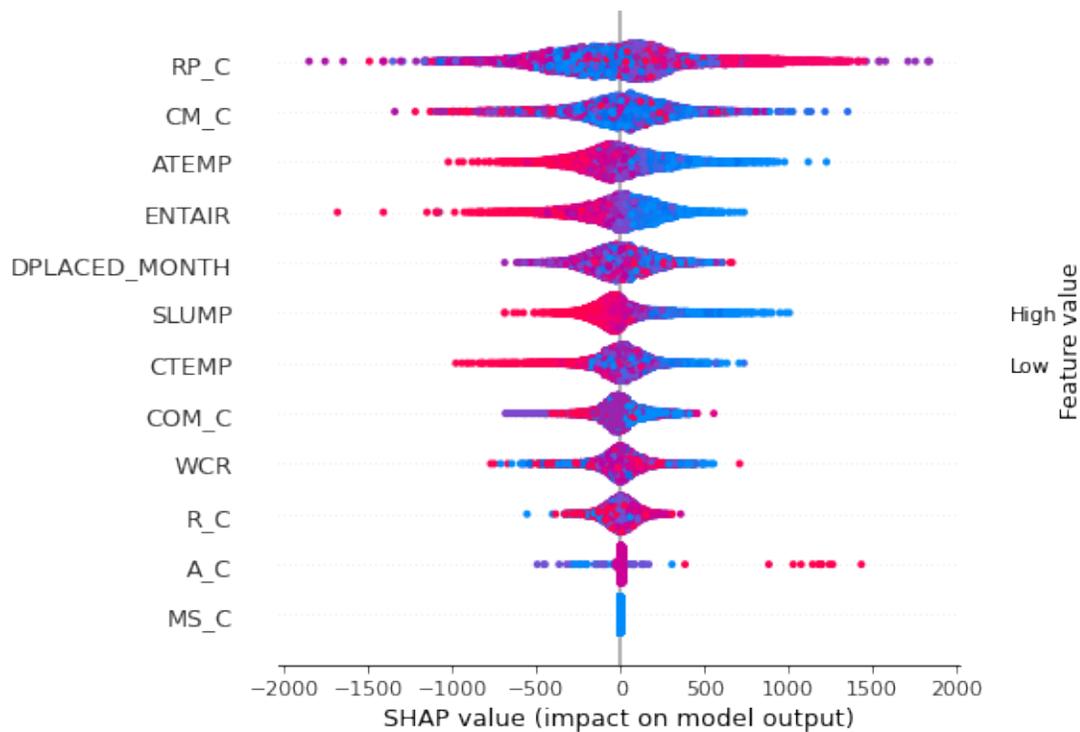


Figure 58. SHAP plot for original dataset, mix 6

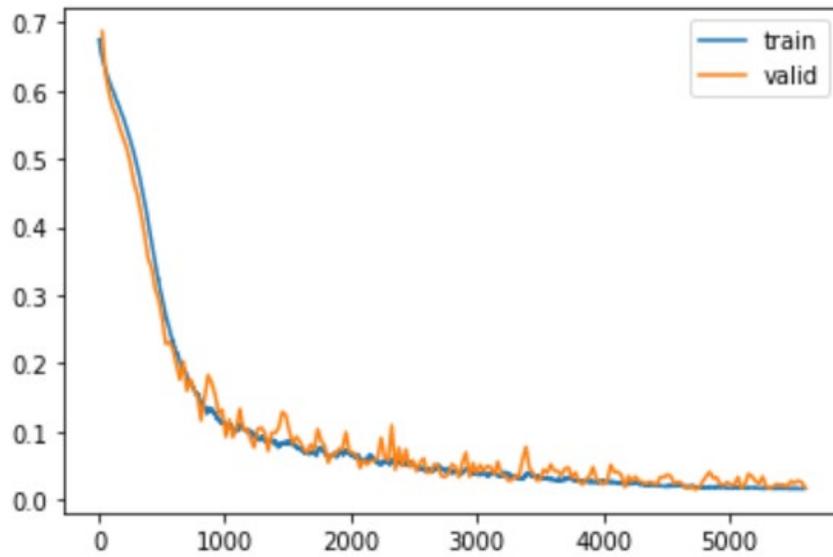


Figure 59 Training Loss Curve for mix 3 binary classification model

	precision	recall	f1-score	support
(-0.001, 3000.0]	0.70	1.00	0.82	228
(3000.0, 8000.0]	1.00	0.99	1.00	18554
accuracy			0.99	18782
macro avg	0.85	1.00	0.91	18782
weighted avg	1.00	0.99	1.00	18782

Figure 60 Accuracy Report for mix 3 binary classification model

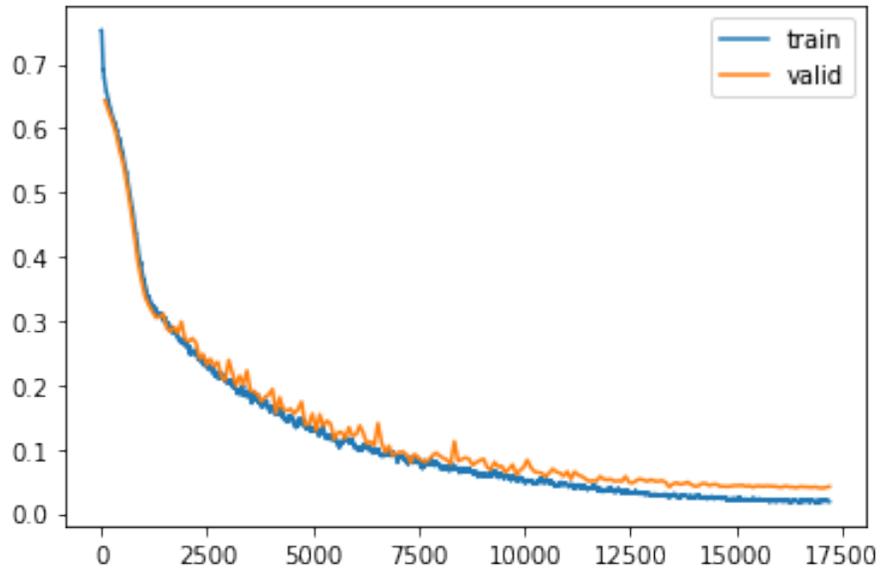


Figure 61 Training Loss Curve for mix 6 binary classification model

	precision	recall	f1-score	support
(-0.001, 4500.0]	0.98	0.99	0.98	997
(4500.0, 10000.0]	1.00	1.00	1.00	7098
accuracy			1.00	8095
macro avg	0.99	0.99	0.99	8095
weighted avg	1.00	1.00	1.00	8095

Figure 62 Accuracy report for mix 6 binary classification model

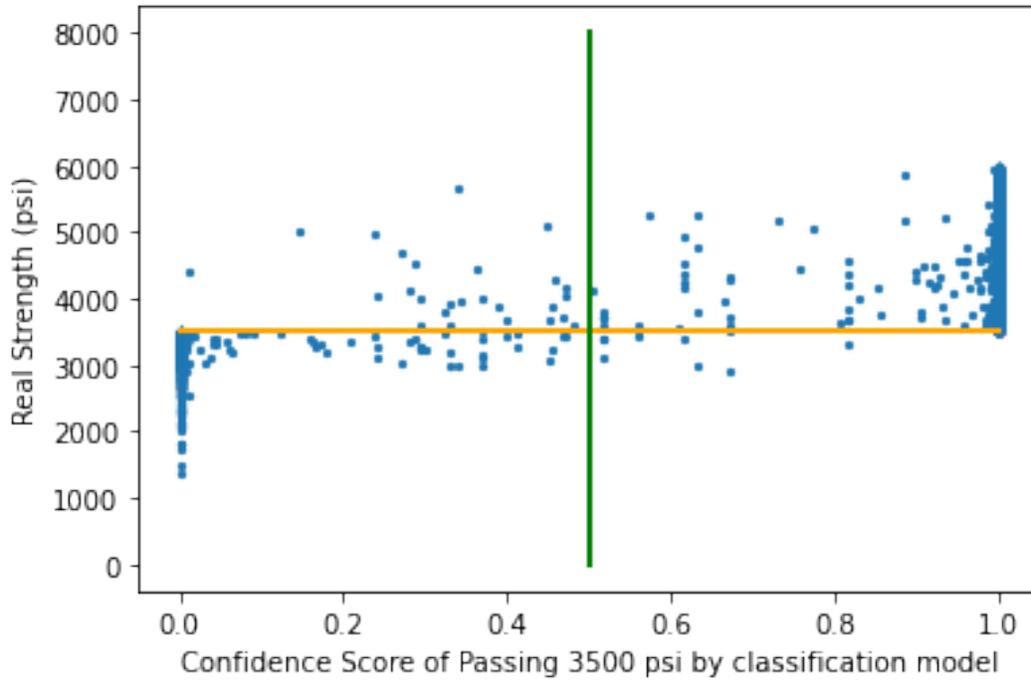


Figure 63 Mix 3 concrete real test strength vs. ML prediction confidence score scatter plot

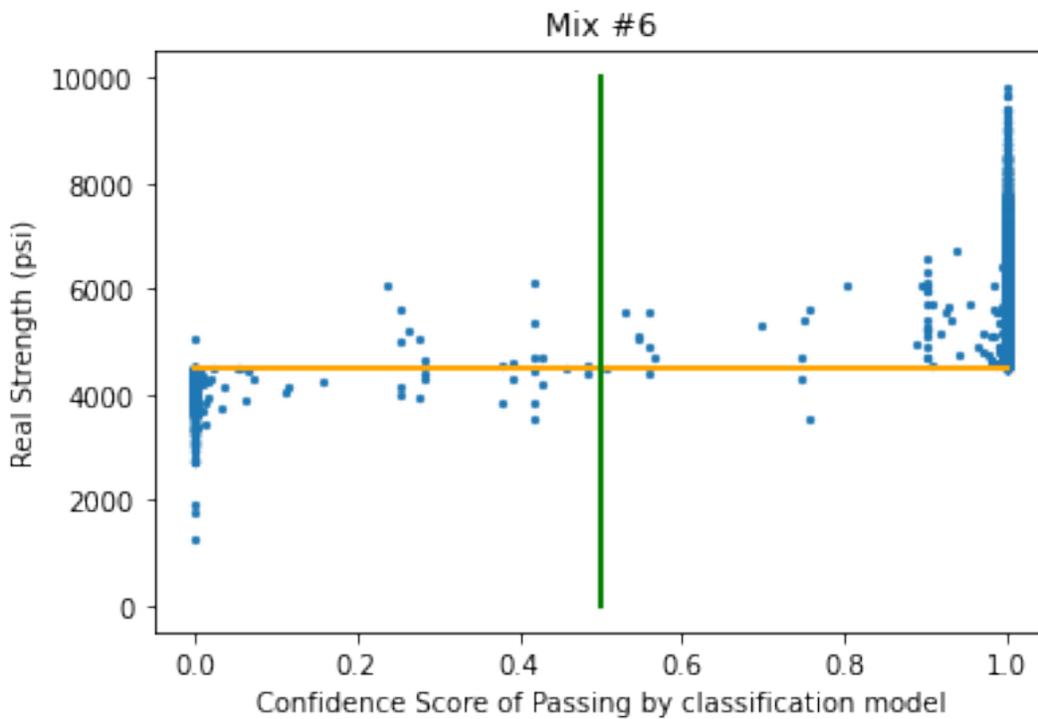


Figure 64 Mix 6 concrete real test strength vs. ML prediction confidence score scatter plot

CHAPTER 6. DEVELOPING NEURAL NETWORK MACHINE LEARNING MODEL FOR PAVEMENT MARKING RETROREFLECTIVITY DATA

The objective of this task is to train neural network models for selected highway tabular dataset of interest to MDOT SHA, specifically pavement marking retroreflectivity data in this chapter, and then use the validated neural network models for predicting the pavement marking retroreflectivity value at given location and age. Plotting the variation of predicted pavement marking retroreflectivity values with their ages would reveal the deterioration relationship of pavement markings. Data preparation including extracting relevant data entries from existing datasets, removing null data, filling missing values, discarding redundant data samples, normalization and converting data into acceptable format by the Fast.ai neural network models were conducted first to make the dataset suitable for machine learning model training.

Pavement markings are vital components of transportation infrastructure, providing valuable, life-saving information to road users. The various forms of markings on road surfaces are critical safety assets with essential safety functions. The information conveyed to the road users by pavement markings are crucial to most short-range and long range driving decisions (Carlson et al. 2013). Transportation agencies in the United States spend millions of dollars every year to apply pavement markings and maintain them in good condition of service. Retroreflectivity of pavement markings is most commonly measured using the Coefficient of Retroreflected Luminance (RL). It is the ratio of luminance (the total amount of light reflected back to the driver) and normal illuminance (the illuminance of the headlights on the marking) and is measured in millicandelas per lux square metre ($\text{mcd}/\text{m}^2/\text{lux}$). There are two standard practices for measuring retroreflectivity: vehicle-mounted retroreflectivity measurement equipment (e.g., Laserlux) that is mostly used for measurement in highways or major multi-lane roadways; and handheld retroreflectometer (e.g., LTL-X) (Sassani et al. 2021). Measured retroreflectivity data considered in this study were taken using Laserlux mobile retroreflectometer, which uses a scanning laser source at an angle, and measures the laser beam reflected back, as defined by ASTM Standard E3320-21. Transportation agencies account for driver preference of retroreflectivity by specifying minimum values ranging between 100 and 150 $\text{mcd}/\text{m}^2/\text{lux}$ as the

minimum. Since retroreflectivity changes (deteriorates) with age, this threshold should be considered when evaluating pavement marking performance for restriping purposes (Hawkins and Smadi 2010).

In developing the pavement marking retroreflectivity machine learning (ML) model, the research team first merged several distinct data sources and created large pavement marking retroreflectivity dataframe for machine learning model training. The original data sources include the following datasets from separate files: application date data from Condition_Data_Worksheet_09-09-2022.xlsx, application date data from 202x FQA Dashboard.xlsx, application date data from JG xx-xx-2x ElectronicMaster.xlsx, Laserlux measured retroreflectivity data and inspection date data from 202x DUMP files. Python script has been written to import the condition data worksheet, the record table in each excel worksheet has been extracted and merged into a large data frame. A python script was also created to calculate the duration days (i.e. age of pavement marking) between inspection date and most recent installation date obtained from the FQA dashboard dataset. Final step includes reading and merging the 'Electronic master' data, updating the nearest neighbor installation date search methodology, merging the laserlux test data with 'condition work sheet', 'Dashboard' and 'Electronic Master' files with county name, route name and mile points. The duration days were then compared with the original ones calculated from condition worksheet, the smallest positive value duration days were then used as the updated duration (in days) feature of the pavement marking at the time of Laserlux based retroreflectivity measurement. The corresponding material was selected as real pavement marking material such as 40-mil paint. Table 8 shows the pavement marking materials considered in this study. Feature extracted from condition data worksheet include the application date material and mile points. The duration for each record was calculated by the inspection date from Laserlux data files and nearest mile marker (i.e. mile points) with application date. Records with missing features were removed. The features from the condition data worksheet were attached to the 1/100th mile Maryland route mile markers based on beginning mile point (BMP) and ending mile point (EMP). In this dataframe with 1288 records, the material, application date (installation date) are included as training features.

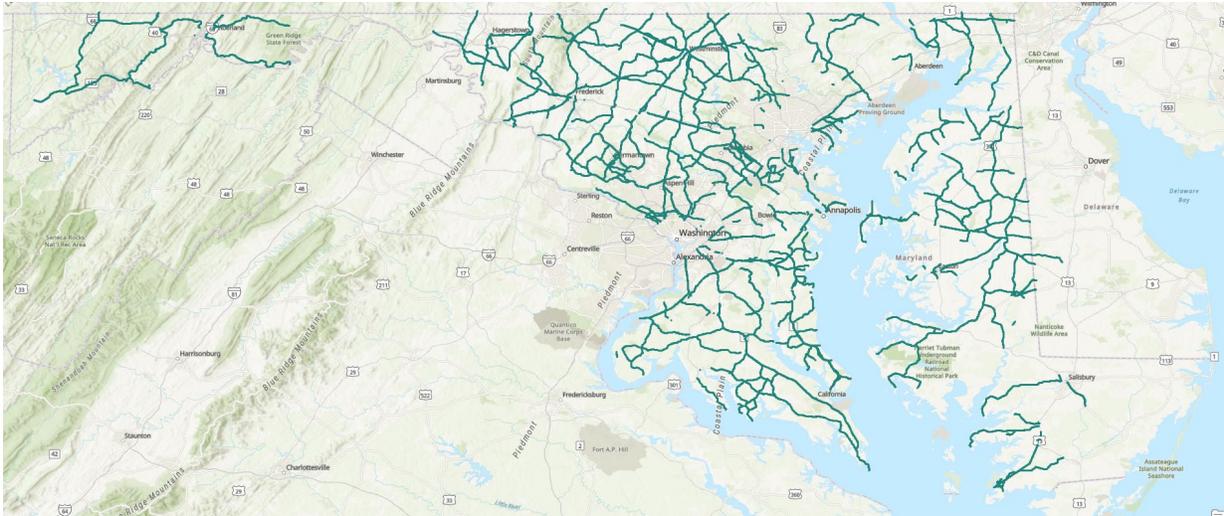


Figure 65. Geospatial visualization of pavement marking retroreflectivity dataset in arc gis pro

For the machine learning model training, the following features are considered as inputs to the model: striping material, age (duration from application date to inspection date, in days), functional class and AADT data to create the new training dataset. Shapefile for the dataframe was generated to visualize the geospatial distribution of pavement marking data in Arcgis pro. The map shown in Figure 65 visualizes the geospatial distribution of Laserlux-measured pavement marking retroreflectivity data by MDOT SHA from Year 2020 to 2022. A preliminary machine learning model for pavement marker retroreflectivity prediction was first trained.

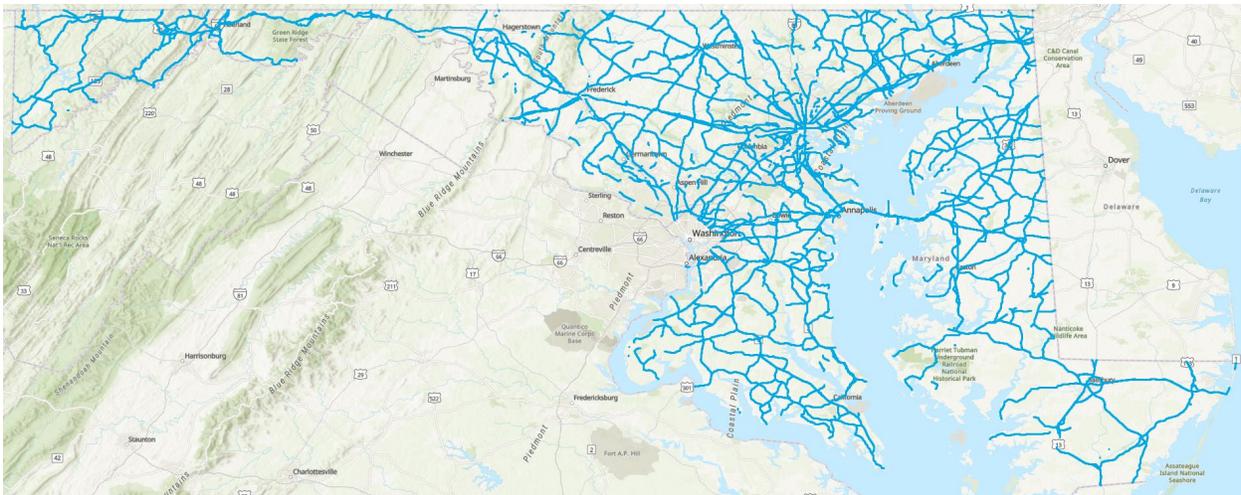


Figure 66. Visualization of 1/100th mile markers with application date (spatial distribution of 1/100 mile markers with application date and material features from condition data work sheet)

In creating the data frame for the pavement marking retroreflectivity data, the application date, material, functional class and AADT were attached to the Laserlux Data record by spatial join with nearest distance search. The searching boundary is restricted within a circle with radius equal to 500 ft.

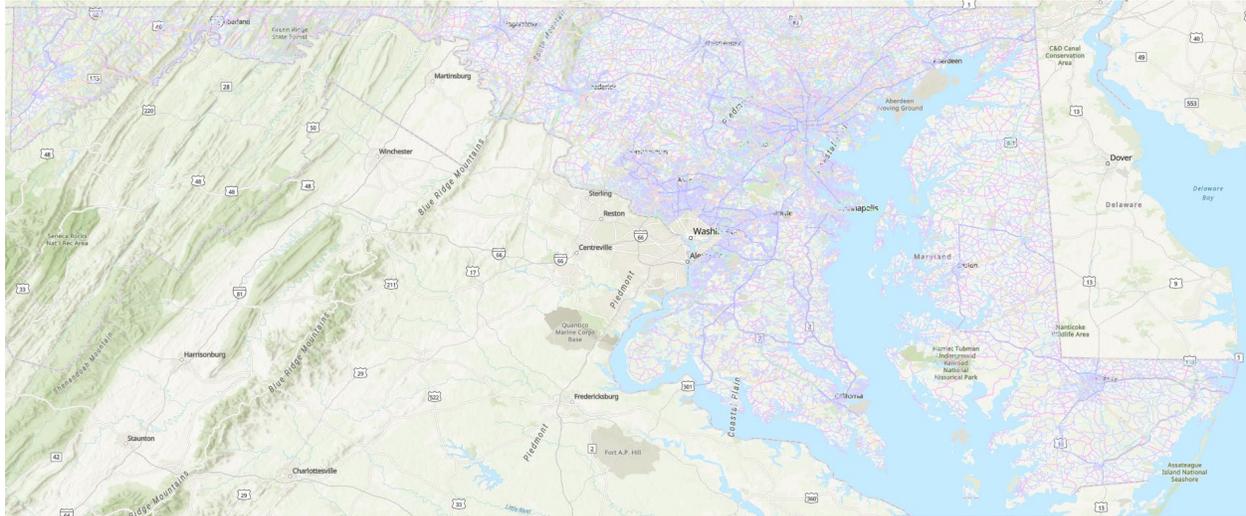


Figure 67. Maryland route network function class map

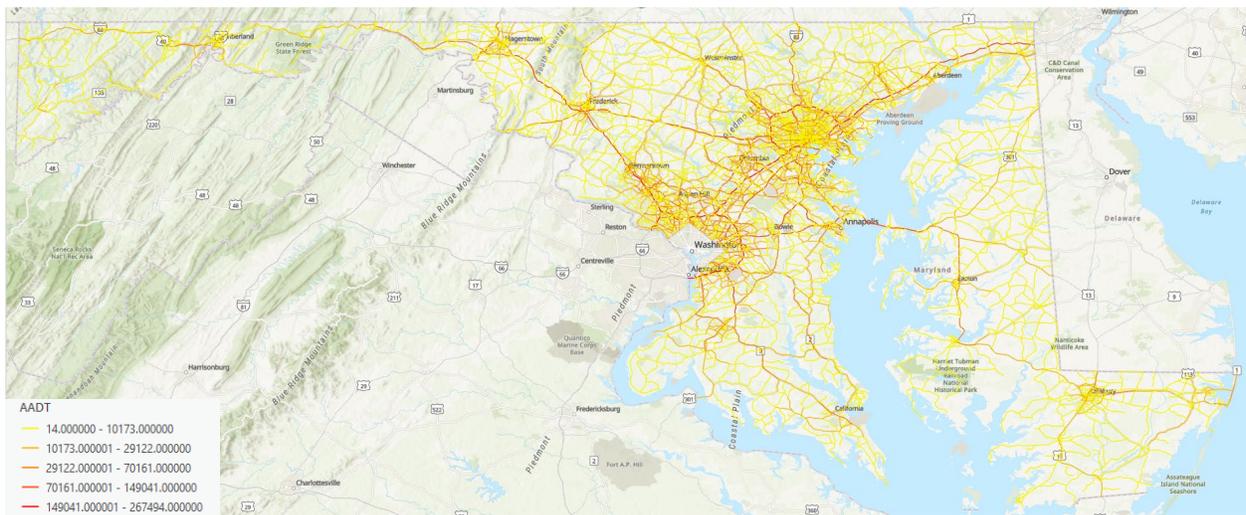


Figure 68. Maryland route network AADT (annual average daily traffic) map

Table 8. Reasonable range of pavement marking age (duration) and retro-reflectivity values

Material	Filter range	Reasonable service life(age) range
40 MIL	White 20-450 & Yellow 20-250	< 2 year
40-Mil Thermo	White 20-450 & Yellow 20-250	< 2 year
60MIL	White 20-450 & Yellow 20-250	< 2 year
90 MIL	White 20-450 & Yellow 20-250	< 2 year
90-Mil Thermo	White 20-450 & Yellow 20-250	< 2 year
EPOXY	White 20-450 & Yellow 20-250	< 2 year
LIQUID THERMO	White 20-450 & Yellow 20-250	< 2 year
PAINT	White 20-350 & Yellow 20-200	< 1 year
PREFORM/ 90MIL	White 20-450 & Yellow 20-200	< 3 year
TAPE	White 20-1650 & Yellow 20-650	< 6 year

Table 8 above lists the reasonable range of duration and retro-reflectivity value given by MDOT SHA engineers. The training dataset was then filtered by applying the reasonable range of duration and retro-reflectivity value. To increase the training dataset size, the feature range (upper bound increased by 150 to mitigate potential artifact effect such as sensor noise) listed in Table 7 was later enlarged for second round machine learning model training. The total number of data samples in the updated training dataset was increased to 17,143 without oversampling.

	Material_j	Application Date_j	Latitude	Longitude	Two_Stripe	StripeType	Road_Condi	FUNCTIONAL	FUNCTION_1	AADT	TRUCK_AADT	Duration
66	90-Mil Thermo	2018-08-15 00:00:00	39.124336	-76.689354	232.5	Single_Solid	0.0	3	Principal Arterial - Other	24082.0	0.0	476
67	90-Mil Thermo	2018-08-15 00:00:00	39.125774	-76.689156	218.4	Single_Solid	0.0	3	Principal Arterial - Other	24082.0	0.0	476
127	40-Mil Thermo	2018-10-29 00:00:00	39.189808	-76.660759	123.1	Single_Solid	0.0	4	Minor Arterial	21711.0	0.0	401
128	40-Mil Thermo	2018-10-29 00:00:00	39.189976	-76.658875	142.7	Single_Solid	0.0	4	Minor Arterial	21711.0	0.0	401
129	40-Mil Thermo	2018-10-29 00:00:00	39.190136	-76.657059	91.6	Single_Solid	0.0	4	Minor Arterial	21711.0	0.0	401
...
115018	90MIL	2019-11-05 00:00:00	39.302574	-76.829849	89.3	Single_Solid	7.0	4	Minor Arterial	18521.0	333.0	1037
115019	90MIL	2019-11-05 00:00:00	39.302616	-76.827957	101.0	Single_Solid	7.0	4	Minor Arterial	18521.0	333.0	1037
115020	90MIL	2019-11-05 00:00:00	39.302586	-76.826103	93.2	Single_Solid	7.0	4	Minor Arterial	18521.0	333.0	1037
115021	90MIL	2019-11-05 00:00:00	39.302608	-76.825638	113.0	Single_Solid	7.0	4	Minor Arterial	18521.0	333.0	1037
115024	90MIL	2019-11-05 00:00:00	39.302654	-76.823723	53.9	Single_Solid	7.0	4	Minor Arterial	14952.0	341.0	1037

85806 rows x 12 columns

Figure 69. Dataframe for pavement marking retroreflectivity machine learning model training

The final dataset for machine learning model training after oversampling includes a total of 85,806 records. Features for training includes 'Material', 'Road Condition', 'Functional class', 'Stripe Color', 'Application month', 'Duration', and 'AADT'. AADT represents annual average daily traffic. The target variable is ‘two stripe average retroreflectivity”, which is the retroreflectivity value measured by Laserlux mobile unit. The feature values were used to filter the training dataset. A total of 9164 out of 85806 training records are kept for training use after filtering. There are 47,469 data samples (out of 115,038 total data samples) with two stripe average retroreflectivity value below 100 mcd/m²/lux, but only data samples with retroreflectivity below 20 were removed from the training set.

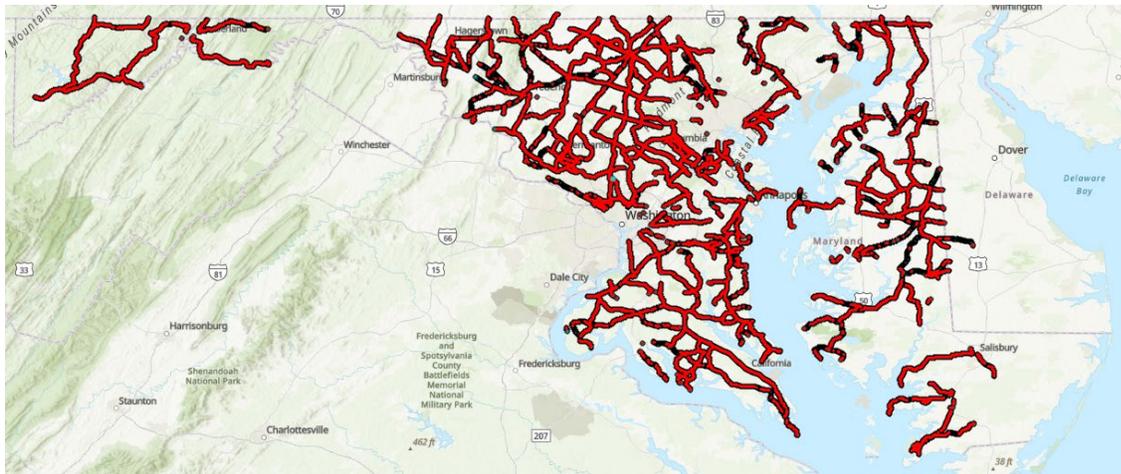


Figure 70. Geospatial distribution of pavement marking retroreflectivity data

Table 9. Duration of pavement marking at retroreflectivity inspection time derived from the three installation-date data sources

APD_Y_db	APD_M_db	APD_D_db	geometry	APP_TIME_DASH	APP_TIME_MASTER	APP_TIME_CDWS	INSP_DATE	DUR-DASH	DUR-CDWS	DUR-MASTER	S_DUR	S_MAT
0.0	0.0	0.0	POINT Z (-76.64175 39.12701 0.00000)	NaT	2019-02-15	2019-02-15	2019-12-04	NaN	292.0	292.0	292.0	90MIL
0.0	0.0	0.0	POINT Z (-76.64253 39.12573 0.00000)	NaT	2019-02-15	2019-02-15	2019-12-04	NaN	292.0	292.0	292.0	90MIL
0.0	0.0	0.0	POINT Z (-76.64331 39.12443 0.00000)	NaT	2019-02-15	2019-02-15	2019-12-04	NaN	292.0	292.0	292.0	90MIL
0.0	0.0	0.0	POINT Z (-76.64413 39.12308 0.00000)	NaT	2019-02-15	2019-02-15	2019-12-04	NaN	292.0	292.0	292.0	90MIL
0.0	0.0	0.0	POINT Z (-76.69379 39.10230 0.00000)	NaT	2020-07-09	2020-07-09	2019-12-04	NaN	-218.0	-218.0	-218.0	PAINT

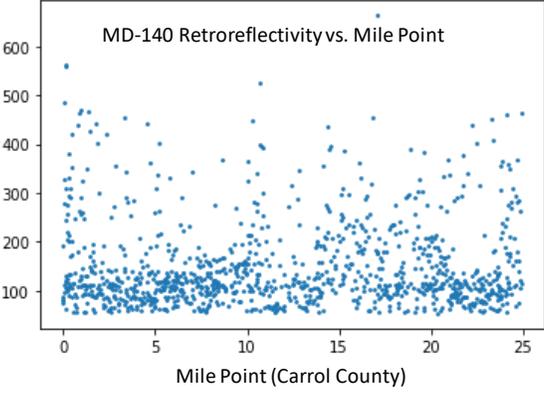
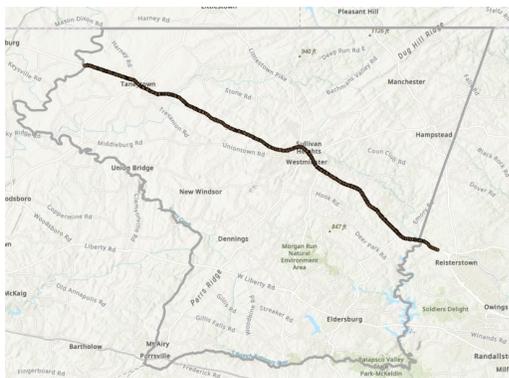


Figure 71. MD-140 pavement marking retroreflectivity field test data matched to 1/100 mile makers

The new (early April) field measurement data is merged to dataframe and visualized in arcgis pro, the data is spatial joined with 1/100 mile markers (i.e. mile points). The MD-140 Retroreflectivity vs. mile points is shown in Figure 71 to visualize the data dispersion.

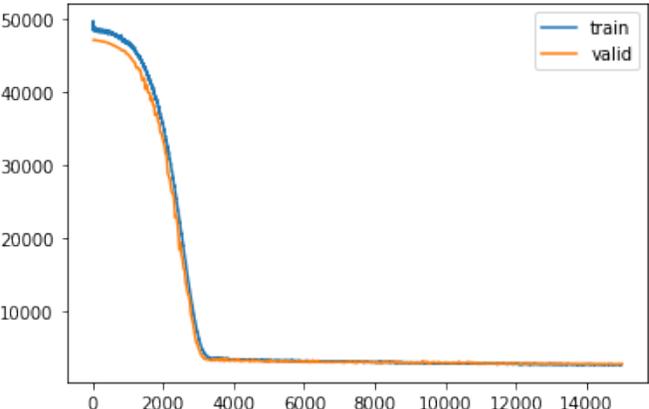
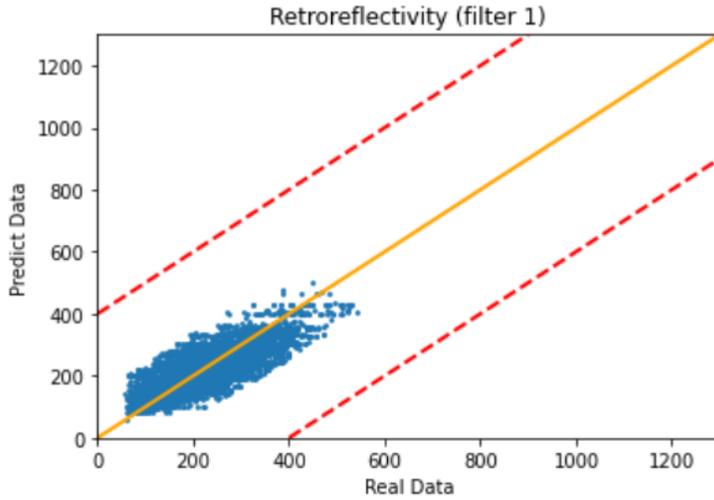


Figure 72. Training loss curve of white 40-mil pavement marking retroreflectivity machine learning model

After new pavement marking retroreflectivity data from the 2020-2022 FQA dashboard dataset became available to the research team and processed in python environment, the retroreflectivity machine learning model was re-trained by appending the new data to the original training dataframe. Over-sampling the combined marking reflectivity dataset was also done for the yellow and white striping colors with the goal to overcome the data imbalance issue in the original dataset. Figure 72 shows the training loss curve of white 40-mil pavement marking retroreflectivity machine learning model, while Figure 73 shows a scatter plot for the updated

machine learning model predictions vs. true white-40mil pavement marking retroreflectivity data. The machine learning model training converged after 500 epochs with R^2 value equal to 0.71.



RMSE: 46.085429779175925
R2SCORE: 0.7058477001283352

Figure 73. Updated pavement marking retroreflectivity machine learning model predictions vs. true value (from field test) for pavement marking of White striping color

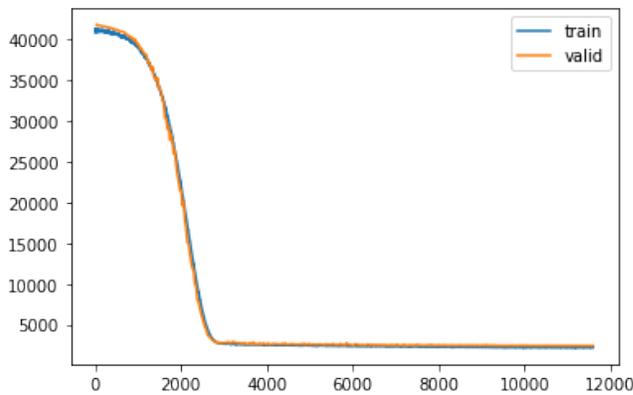
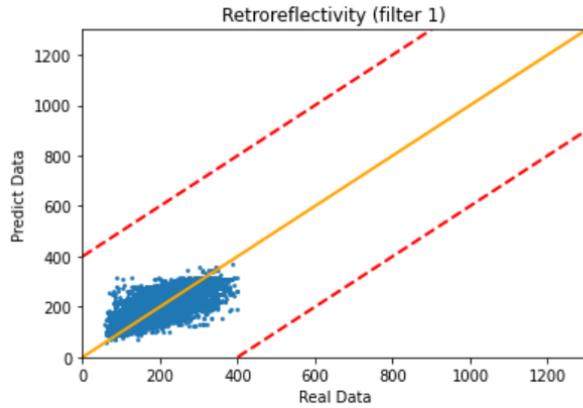


Figure 74. Training loss curve of machine learning model using Yellow 40-mil pavement marking retroreflectivity data



RMSE: 45.76599240528728
R2SCORE: 0.6090071563407448

Figure 75. Updated pavement marking retroreflectivity machine learning model predictions vs. true value (from field test) for pavement marking of Yellow striping color

The tabular machine learning model for Yellow striping color pavement marking retroreflectivity is updated by addressing the data imbalance issue through oversampling. Figure 74 shows the training loss curve of Yellow 40-mil pavement marking retroreflectivity machine learning model, while Figure 75 shows a scatter plot for the updated machine learning model predictions vs. true Yellow 40-mil pavement marking retroreflectivity data. The machine learning model training converged after 500 epochs with R^2 value equal to 0.61.

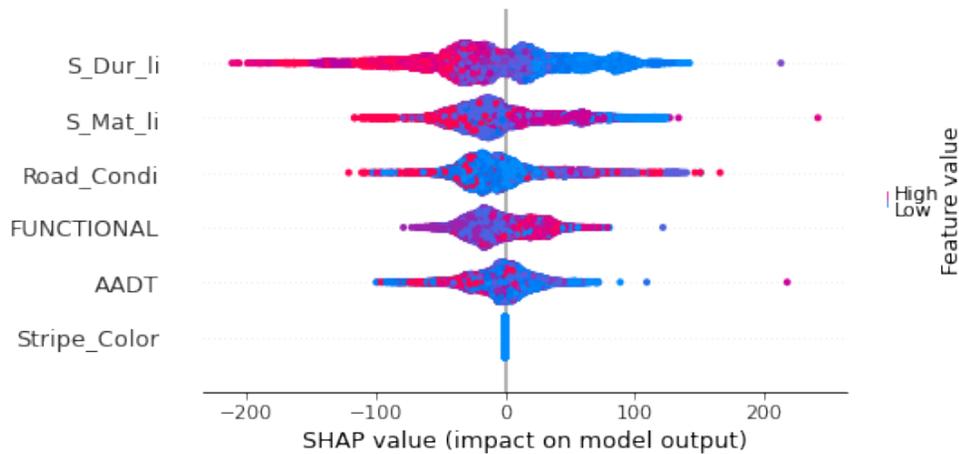


Figure 76. Feature importance ranking for white 40-mil pavement marking retroreflectivity machine learning model

The tabular machine learning model for pavement marking retroreflectivity is evaluated through a feature importance study with SHAP analysis was also performed. The SHAP plot for white color pavement marking retroreflectivity model suggests the duration and material type are the two most influential factors.

CHAPTER 7. DEVELOPING NEURAL NETWORK MACHINE LEARNING MODEL FOR GEOTECHNICAL DRILLING DATA

The objective of this task was to update and modify the neural network models developed for various geotechnical drilling datasets including grain size, groundwater table depth, SPT N-Value ranges. Phase 2 research further developed the model from Phase 1 by including more data and feature variables. Data resampling techniques were also applied to the dataset to improve the performance of models. The resulting model is fully implemented and is utilized as part of the soil boring request process in MDOT SHA to deliver preliminary subsurface data based on machine learning analysis with over 10,000 borings in a centralized database, and then use the validated neural network models for dependable variable prediction. Data preparation including extracting relevant data entries from existing datasets, removing null data, filling missing values, discarding redundant data samples, normalization and converting data into acceptable format by the Fast.ai neural network models have been conducted.

In this study, the research team reviewed drilling (SPT) data based machine learning model training and predictions, and investigated optimization approach to the drilling data models; The research team also did literature review to ensure the tabular data neural network model is current and has been updated for the newly available drilling data.

Two datasets were used for machine learning model training to predict the grain size category at specific drilling depth. The first machine learning model was trained by a total of 293,728 data samples from MDOT SHA drilling sites. Feature variables including elevation, drilling depth. Latitude and longitude have been extracted from the MDOT SHA database. A nearest neighbor search was employed to extract the major rock type from the USGS lithology map. Samples with missing features have been removed and a total of 285,944 data samples are available for machine learning model training. The second model was trained by selected data from Maryland and consolidated with additional drilling database from Ohio and Missouri. The pre-processed data includes a total of 248,686 data samples for training. The distribution histogram for both datasets are shown in Figures 77 and 80, showing that sand and silt are the two major types of soils in the Maryland drilling data and each of these two soil type data has over 100,000 data

samples. Gravel has the fewest data samples in the training dataset, with less than 2,000 records. For the consolidated dataset, the majority category is Clay, which has over 120,000 data samples. The minority soil type is also Gravel, which has fewer than 10,000 samples. The spatial distribution of grain size categories is visualized in Figures 78 to 79. The grain size in mid and east regions of Maryland at shallow drilling depth are mostly sand while for west region it is silt. Data is concentrated mostly at relatively shallower depth in the west region of Maryland than the larger depth in east region.

There are also two datasets used to predict the ground water depth through machine learning model. The first machine learning model was trained with a total of 7,760 testing data from Maryland drilling project. Data cleansing and preprocessing was applied to the dataset to remove samples with missing feature variable values and target variables. A total of 6,984 data samples were included for training. The other model was built with data extracted from USGS nationwide database. The USGS dataset includes 9,673 training samples after data preprocessing and cleansing. The distribution histogram was plotted in Figures 85 to 86. The histograms suggest that most data samples have ground water depth less than 10 ft, and the most populous water-depth bucket is 2-4 ft for Maryland data which has over 1,000 samples, and USGS nationwide data, which has over 2,000 samples.

Tabular machine learning models have also been developed to predict the Standard Penetration test (SPT-N) test values. The SPT-N value provides information regarding the soil strength. For example, SPT-N value may be a meaningful indicator of the friction angle of sandy soils and clay soils and the stiffness of the clay (Ruwan 2016). The SPT-N dataset includes a total of 37,457 samples available for machine learning model training. Features variables for predicting the SPT-N number buckets include GPS coordinates, rock type, drilling month, drilling depth and elevation. Samples with missing feature variable values and target variables were discarded from the training dataset. The histogram in Figures 82 visualized the data distribution of SPT-N dataset. The refusal depth test is defined as the depth or point which the hammer does not advance for >50 blows/6 inches. The histogram of refusal depth bucket is shown in Figure 88. The bucket of 21-50 has a peak distribution among other buckets in terms of amount of data. The

spatial distribution has also been visualized. Mid and west regions have most data in the 50+ bucket.

Parametric study has been done to select feature variables and the optimal machine learning model hyper-parameter values. Multiple-label classification machine learning models were built for the Maryland and consolidated grain size dataset. Since both models were trained with dataset containing over 200,000 data samples, both models used a batch size of 2,048. The hidden layers include two 200-nodes layers to connect the input layer and output layer in the neural network of model. Model 1 was trained for 40 epochs while Model 2 was trained for 30 epochs to ensure they are converged without overfitting or under fitting issue in training. The accuracy report for the machine learning predicted dataset exhibits an average f1-score of 0.65 for all categories. The preliminary result shows the minority category of gravel has a lower f1-score than other categories. The scatter plot of real category vs. predicted category is visualized in Figure 91. The radius of the circle represents the size of dataset (numbers of samples) falling inside the corresponding soil category.

Multi-label classification models were also trained for the SPTN dataset. The SPTN value was divided into 11 buckets and the model was trained to predict the corresponding bucket of the SPTN value. The input feature variables include the GPS coordinate and elevation of the drilling site, the drilling month, the USGS major rock type, and the drilling depth. The batch size for both models was taken as 128. The hidden layers include three 200-neurons layers. The training epoch for each model is 40. The accuracy reports for both models The average accuracy is similar for the two models, both are around 0.4.

Two regression machine learning models have been developed for the dataset of MD groundwater depth and the USGS groundwater depth. These preliminary machine learning models apply a two-layer hidden neuron network, each layer has 200 neurons. The model was trained for 200 epochs based on the findings from a previously done parametric study and used a batch size of 512. The input feature variables include the GPS coordinates and elevation of the drilling site, the drilling month and USGS major rock type. The prediction test was done on the

Maryland dataset and real data vs. predicted data scatter plot was shown in Figures 95 and 97. The prediction for Maryland dataset reached an RMSE of 6.62 and a R^2 score of 0.56.

The drilling datasets were observed to have imbalanced distribution pattern in the target variable. Therefore, data resampling was applied to improve the model performance by making the distribution relatively more uniform. The dataset of grain size has imbalanced distribution pattern in the target variable since the gravel category has fewer than 2,000 samples while sand and silt categories have over 100,000 samples. Therefore, the data resampling strategy was applied to the grain size dataset. Four machine learning models were trained. These models oversampled the gravel category by 3, 4, 10, 15, 20 times, and two distribution histograms of resampled data are plotted in Figures 102 to 103. It was found that the models duplicated the gravel category by 3-4 times have better average accuracy than other models in machine learning prediction. The oversampled model improved the average accuracy from 0.66 to 0.72, while all other models with larger over-sampled data size have lower average accuracy than the original model. The scatter plot of real category vs machine learning predicted category can also be used visualize the improvement in predicting the minority group.

The data resampling strategy has also been applied to the SPT-N bucket and groundwater depth dataset. Histograms of SPT-N buckets is shown in Figure 105, and data in the 50+ bucket has been down-sampled and data in other buckets have been oversampled. The neural network machine learning model used the same hyper-parameter values as the original model. A prediction test has been done on the original dataset. The accuracy report and the scatter plot of machine learning predicted data vs. real data shows that the average accuracy has improved from 0.4 to 0.51, and the f-1 score for the minority groups have been improved by adopting the data resampling strategy for the retrained model. The ground water depth dataset has also been resampled and the updated machine learning model improved the RMSE to 5.90 and the R^2 score to 0.65.

SHAP analysis was conducted to study the relative contribution significance of various feature variables considered in the training of the drilling dataset machine learning model. The SHAP value plot for the grain size classification model suggests the USGS major rock type is an

important influencing factor for both the Maryland dataset and consolidated dataset. GPS location is also an important influencing factor for the model. For SPT-N bucket classification model, the drilling depth is the most important contributing factor, while latitude and rock type are relatively less important factors for the machine learning model. As visualized in the SHAP value scatter plot for the ground water depth, the drilling site GPS coordinates and elevation are predicted to be the two most influencing factors in machine learning model predictions.

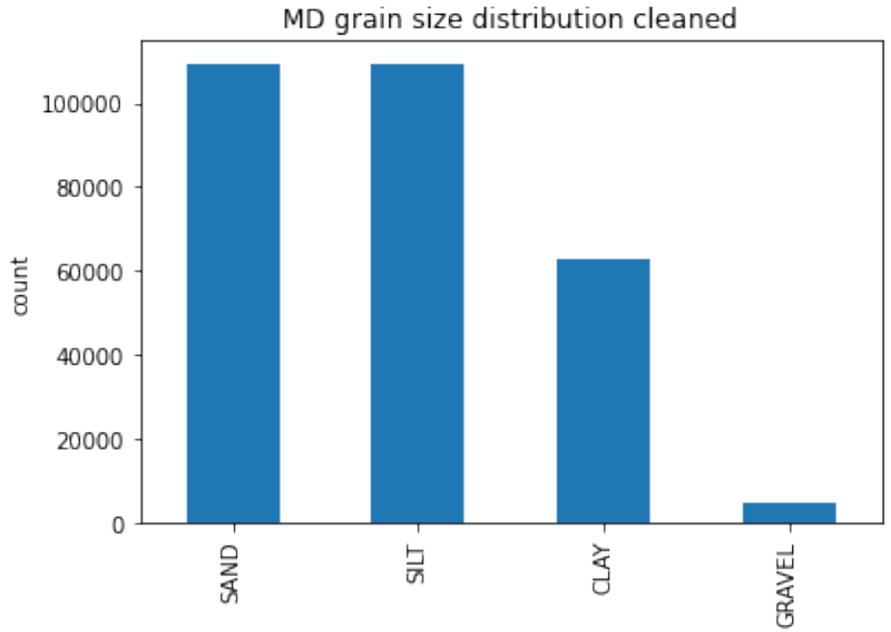


Figure 77. MD grain size dataset distribution

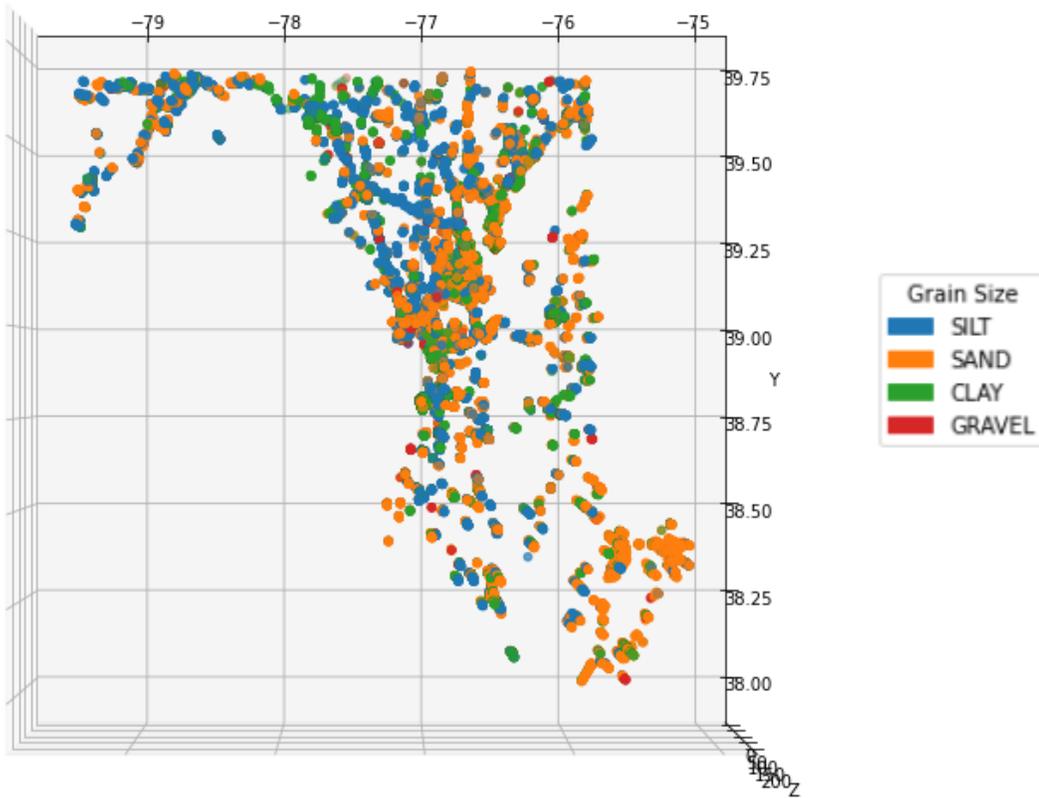


Figure 78. Spatial distribution of Grain Size data (x-y plane)

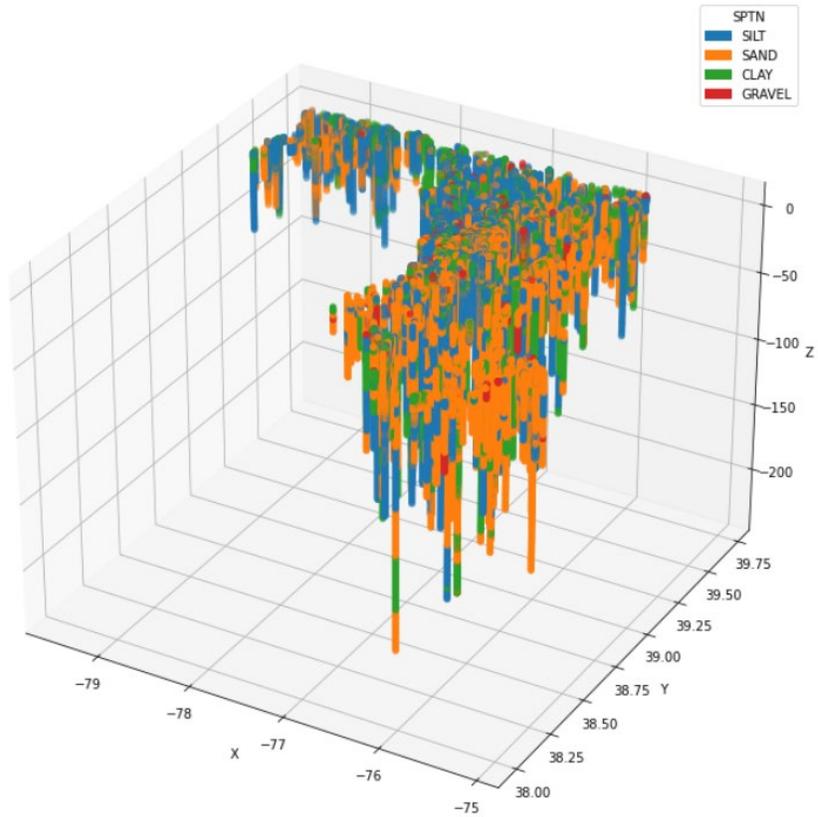


Figure 79. 3D spatial distribution of MD Grain Size Dataset

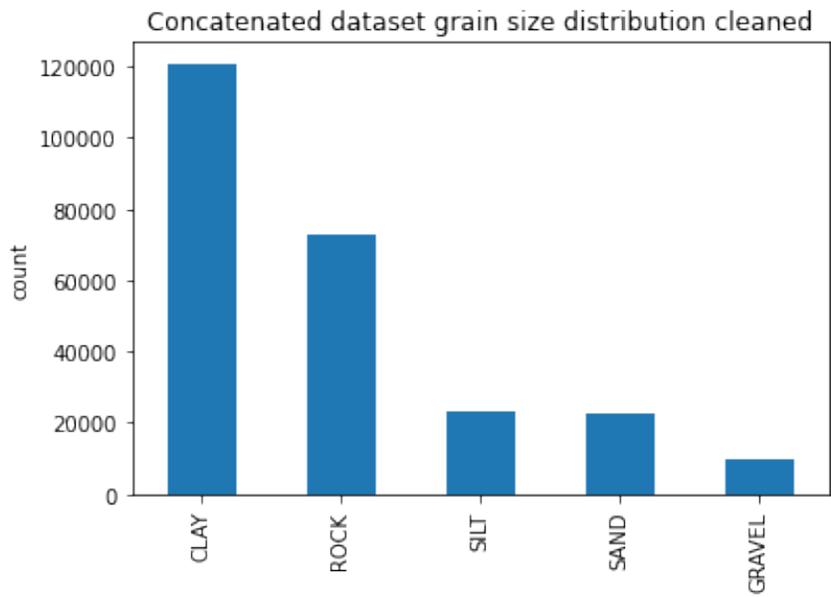


Figure 80. Concatenated grain size dataset distribution



Figure 81. SPT number buckets distribution

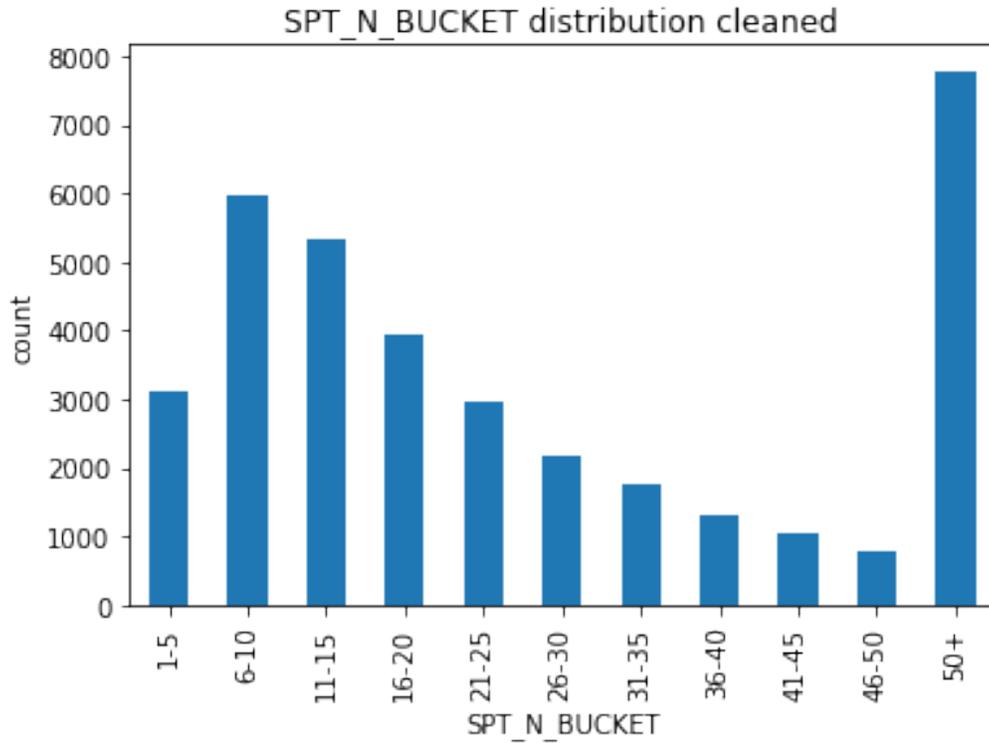


Figure 82. cleansed SPT number buckets distribution

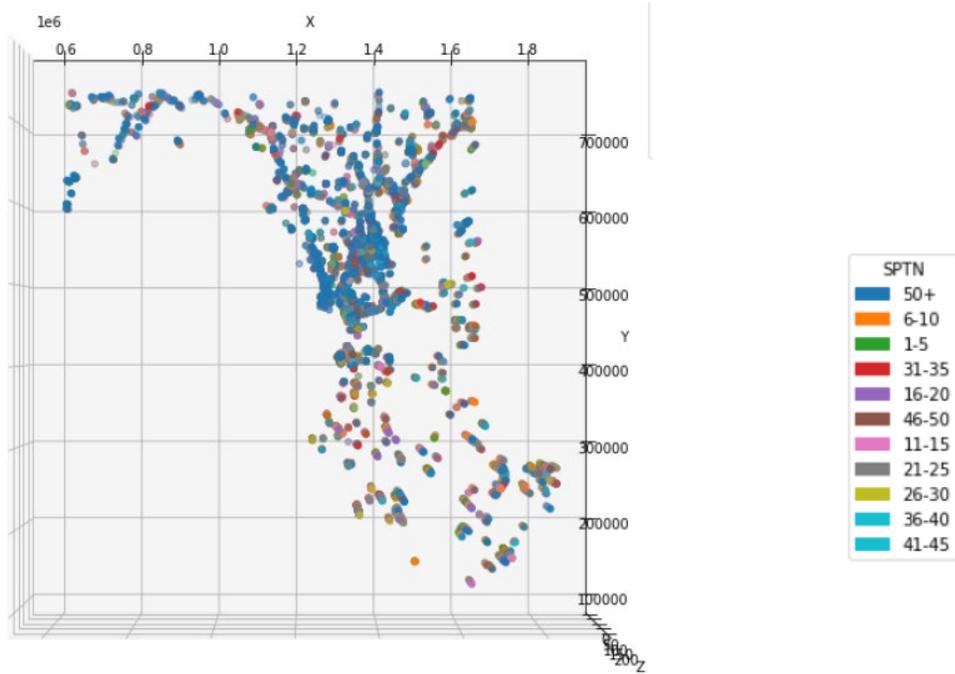


Figure 83. Spatial distribution of SPTN bucket data (x-y plane)

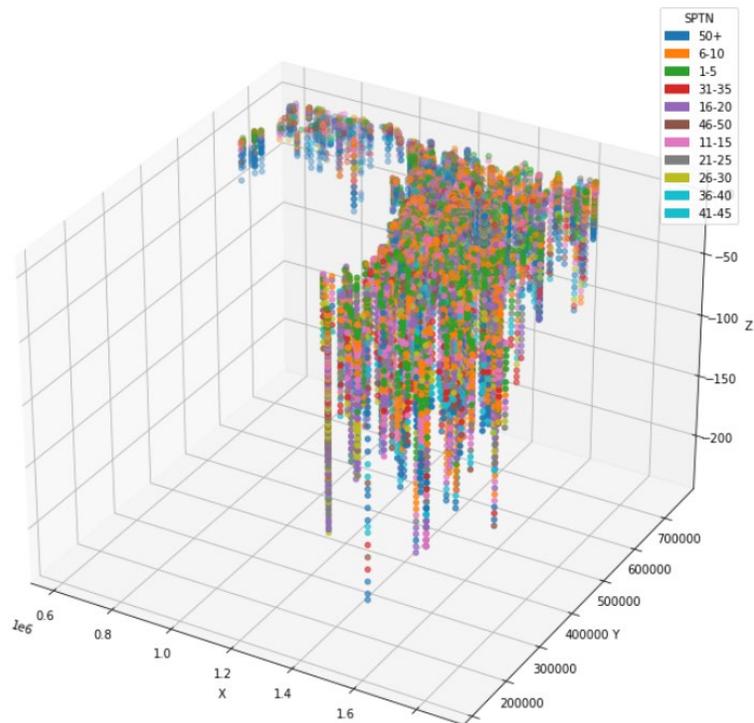


Figure 84. 3D spatial distribution of MD SPTN bucket Dataset

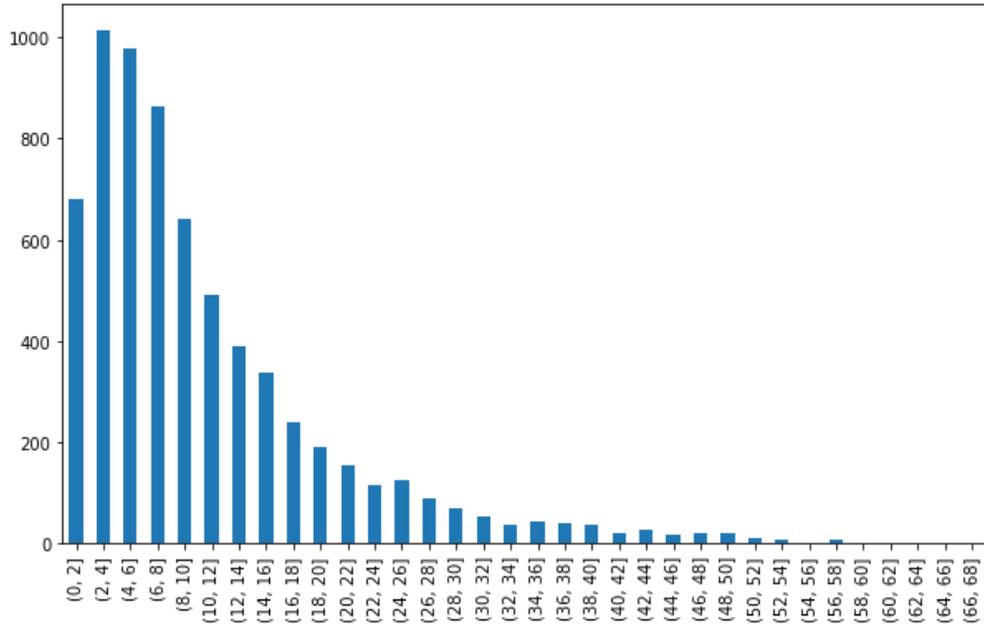


Figure 85. Histogram of MD ground water depth distribution

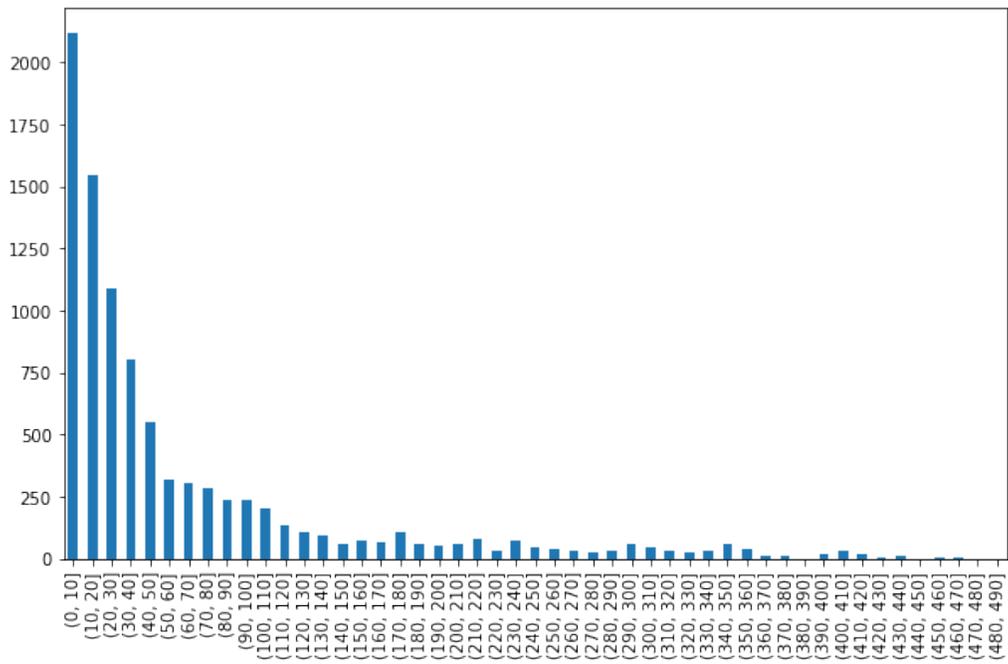


Figure 86. Histogram of USGS nationwide ground water depth distribution

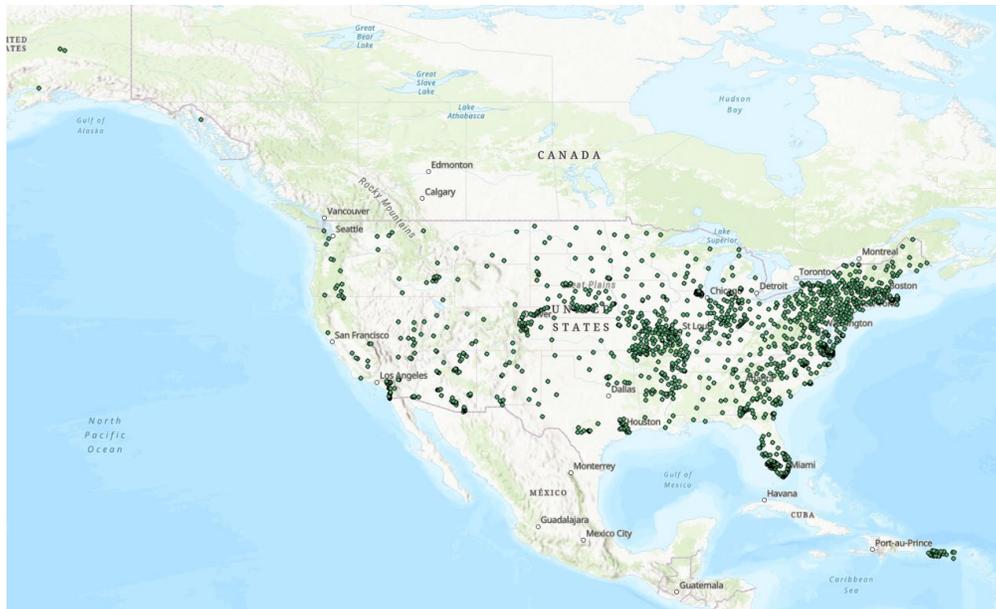


Figure 87. Spatial distribution of USGS nationwide ground water depth distribution

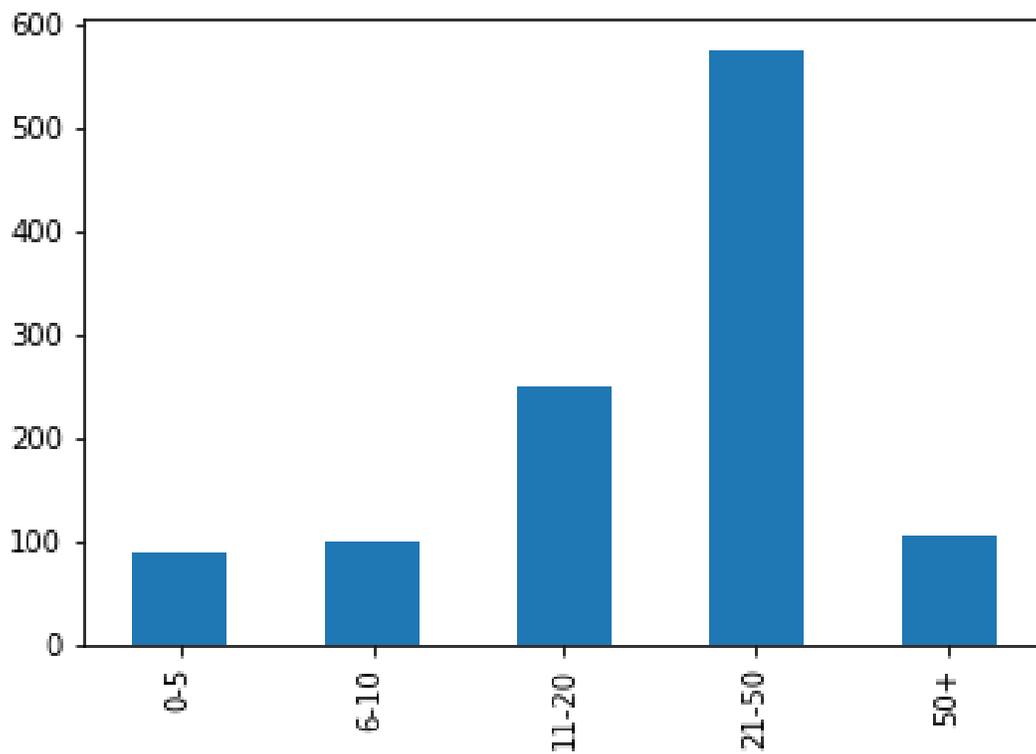


Figure 88. Refusal depth bucket Distribution

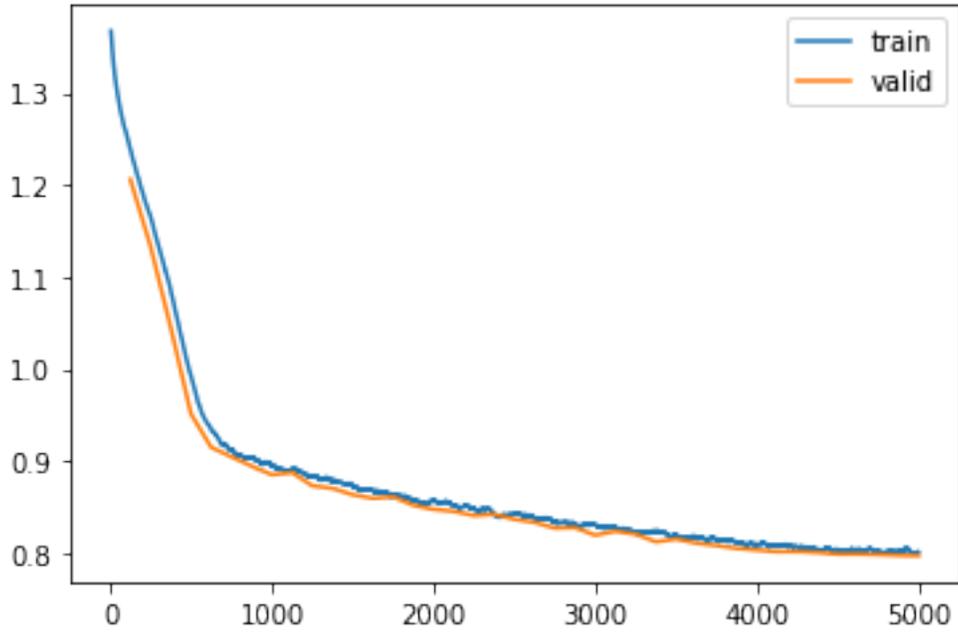


Figure 89. Training loss curve for MD grain size model

	precision	recall	f1-score	support
CLAY	0.64	0.48	0.55	6437
GRAVEL	0.42	0.05	0.08	458
SAND	0.63	0.74	0.68	10831
SILT	0.68	0.68	0.68	10869
accuracy			0.65	28595
macro avg	0.59	0.49	0.50	28595
weighted avg	0.65	0.65	0.64	28595

Figure 90 Accuracy report for MD grain size model

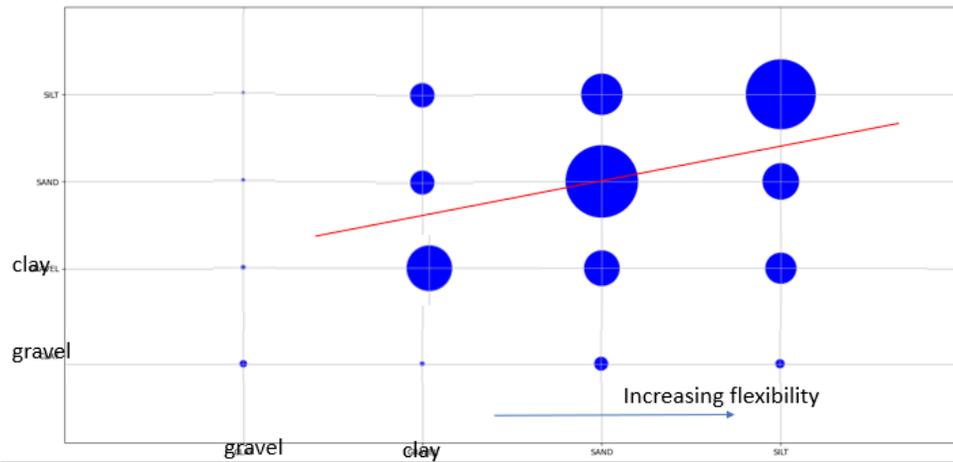


Figure 91. Grain size predicted category - real category scatter plot

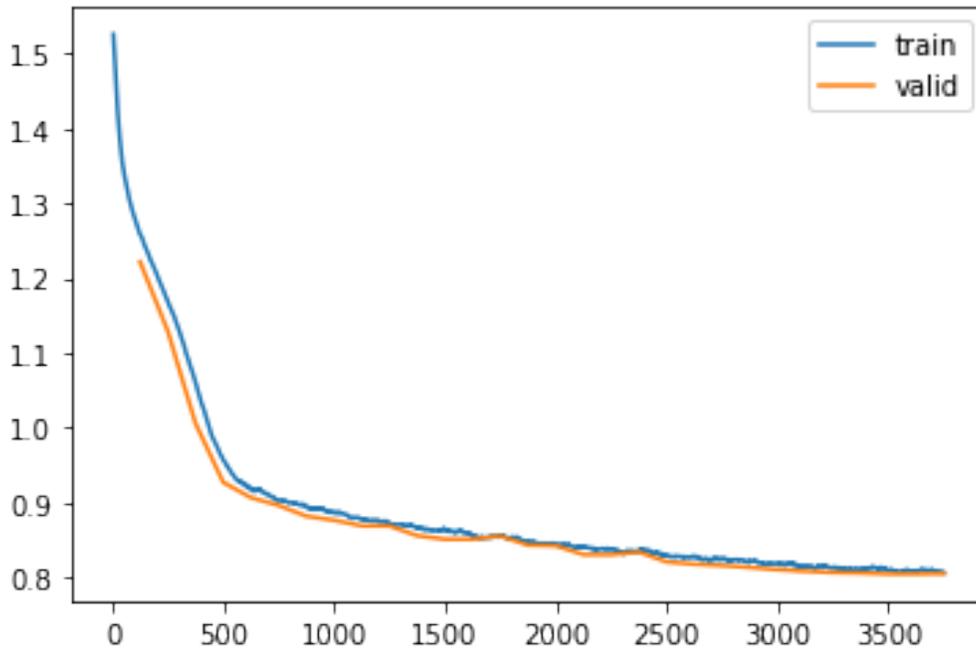


Figure 92. Training loss curve of consolidated grain size data model

	precision	recall	f1-score	support
CLAY	0.62	0.46	0.53	6437
GRAVEL	0.34	0.04	0.07	458
SAND	0.62	0.72	0.67	10831
SILT	0.67	0.69	0.68	10869
accuracy			0.64	28595
macro avg	0.56	0.48	0.49	28595
weighted avg	0.63	0.64	0.63	28595

Figure 93. Accuracy report of consolidated grain size data model

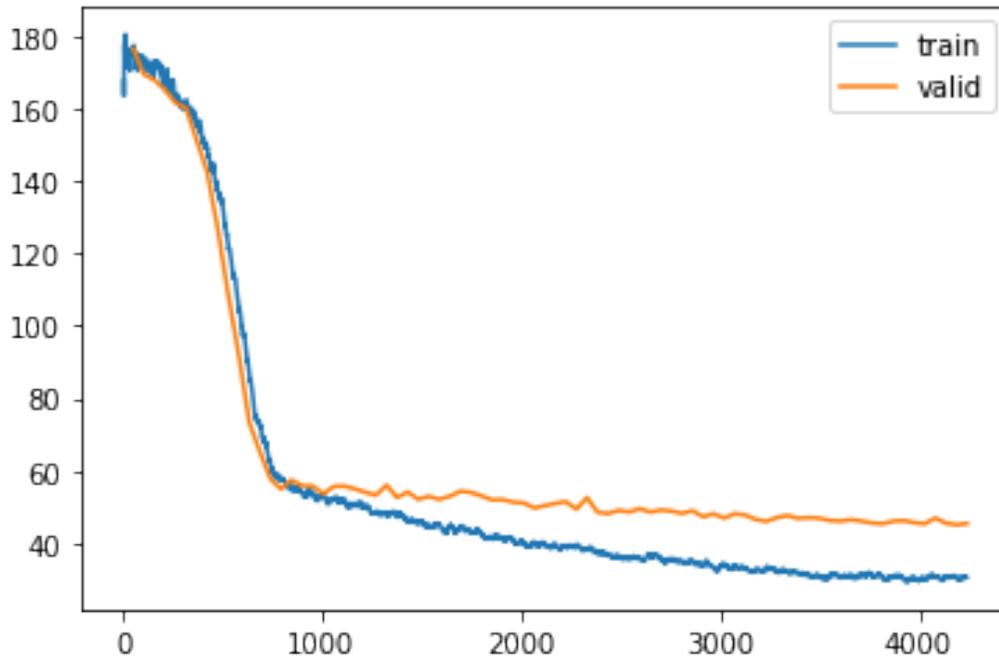


Figure 94. Training loss curve of MD groundwater depth dataset

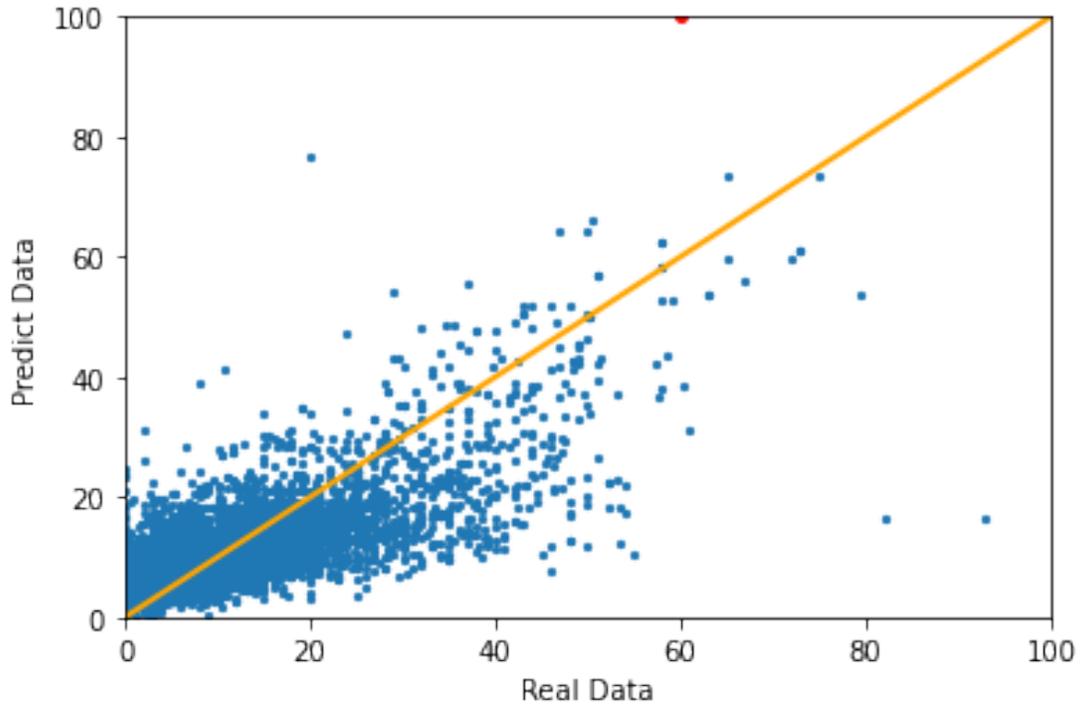


Figure 95. x-y scatter plot of MD groundwater depth model

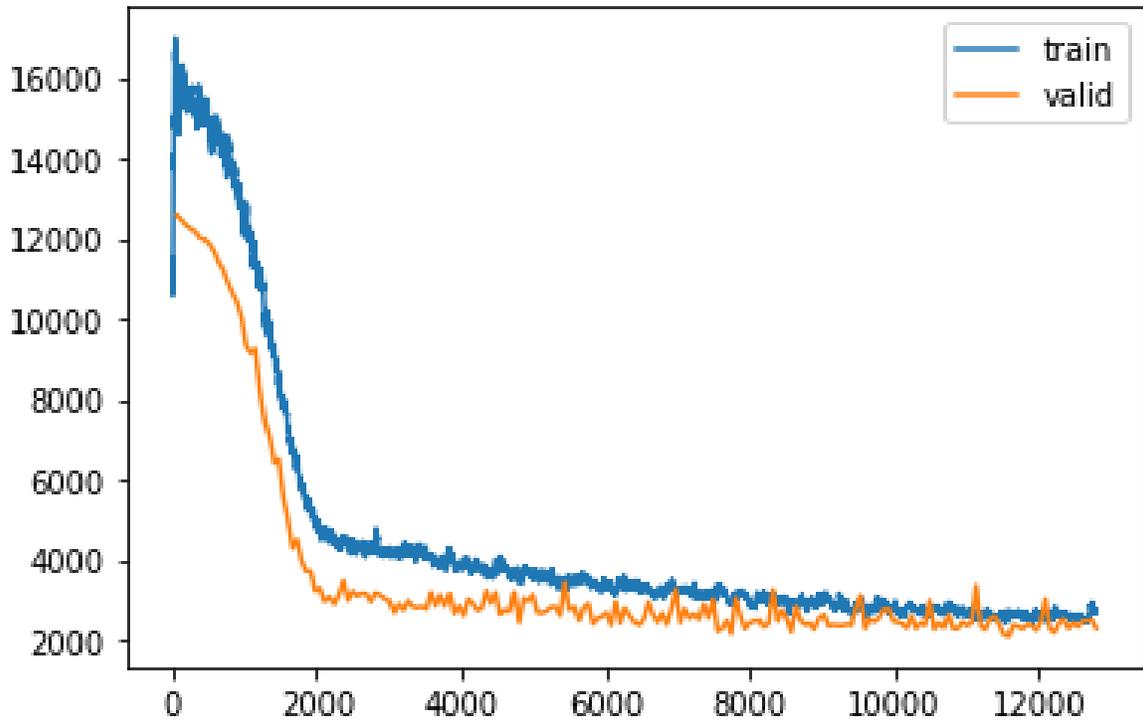
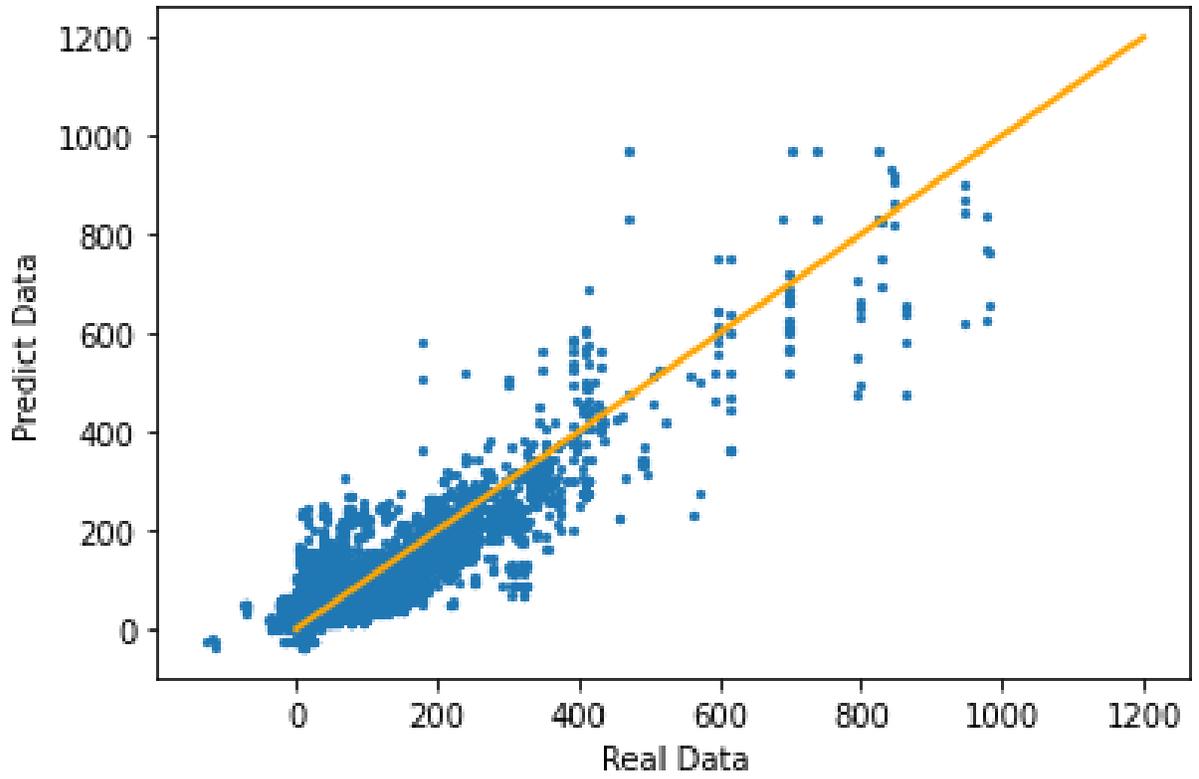


Figure 96. Training loss curve of USGS nationwide groundwater depth model



1

Figure 97. x-y scatter plot of USGS nationwide ground water depth model

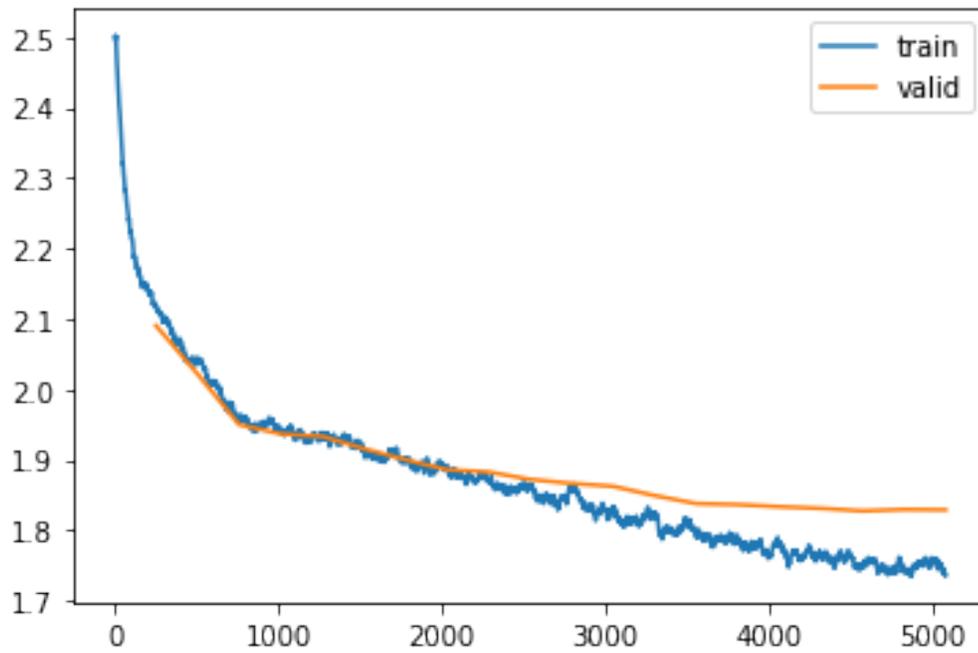


Figure 98. Training loss curve of MD SPTN model

	precision	recall	f1-score	support
1-5	0.44	0.37	0.40	3101
11-15	0.31	0.34	0.32	5325
16-20	0.28	0.16	0.21	3955
21-25	0.28	0.12	0.17	2968
26-30	0.28	0.06	0.10	2161
31-35	0.22	0.08	0.11	1767
36-40	0.23	0.03	0.06	1312
41-45	0.21	0.01	0.02	1038
46-50	0.37	0.01	0.02	781
50+	0.50	0.85	0.63	7786
6-10	0.36	0.57	0.44	5987
accuracy			0.40	36181
macro avg	0.32	0.24	0.23	36181
weighted avg	0.35	0.40	0.34	36181

Figure 99. Accuracy report of MD SPTN model

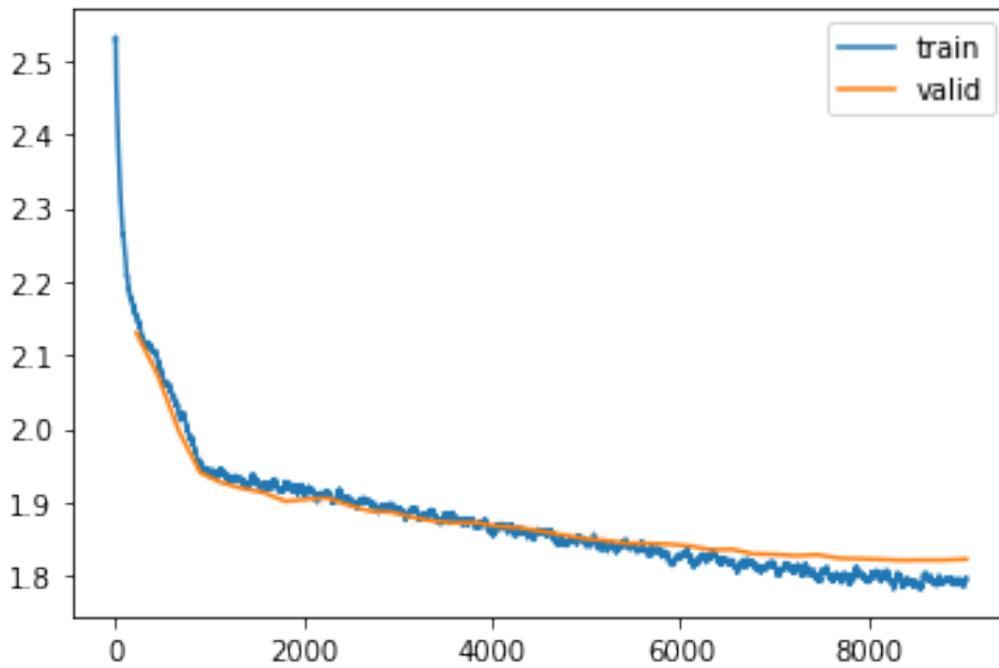


Figure 100. Training loss curve of consolidated SPTN model

	precision	recall	f1-score	support
1-5	0.36	0.32	0.34	3101
11-15	0.26	0.29	0.27	5325
16-20	0.26	0.14	0.18	3955
21-25	0.23	0.09	0.13	2968
26-30	0.28	0.03	0.06	2161
31-35	0.23	0.05	0.09	1767
36-40	0.00	0.00	0.00	1312
41-45	0.00	0.00	0.00	1038
46-50	0.00	0.00	0.00	781
50+	0.47	0.82	0.60	7786
6-10	0.33	0.56	0.42	5987
accuracy			0.37	36181
macro avg	0.22	0.21	0.19	36181
weighted avg	0.30	0.37	0.30	36181

Figure 101. Accuracy report of consolidated SPTN model

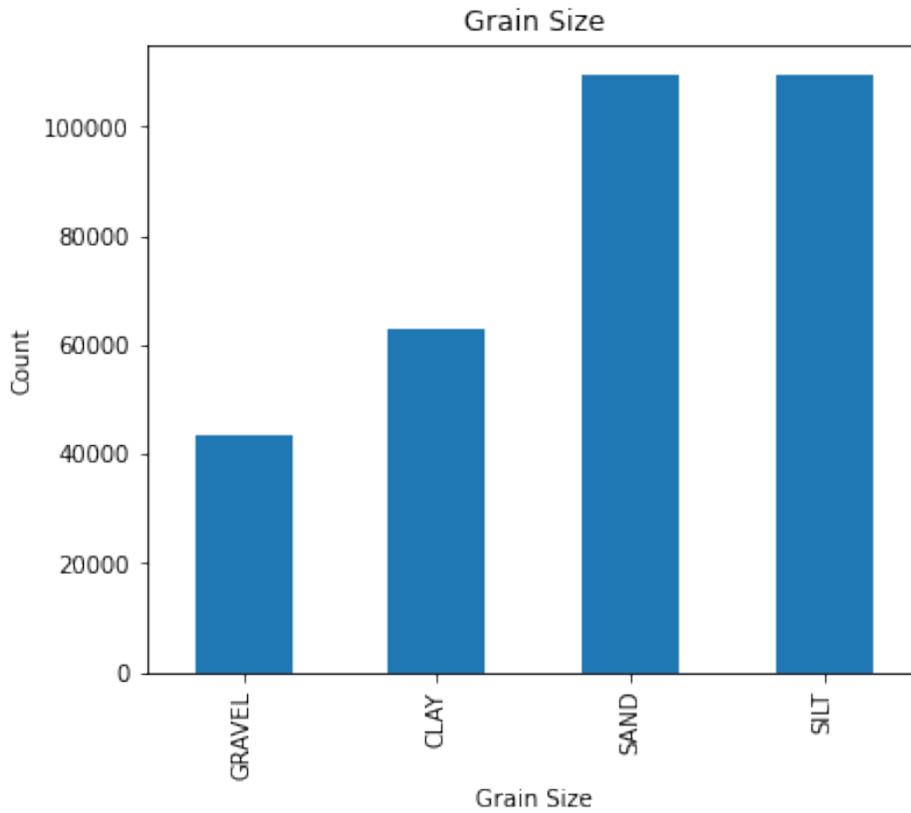


Figure 102. Histogram of the resampled MD grain size dataset 1

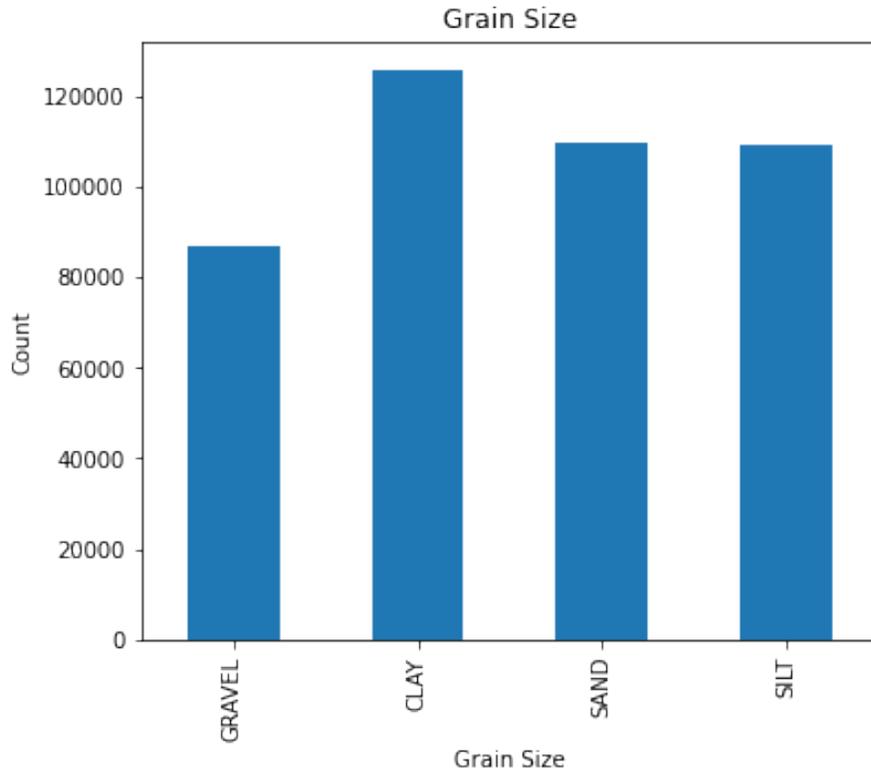


Figure 103. Histogram of the resampled MD grain size dataset1

	precision	recall	f1-score	support
GRAVEL	0.37	0.52	0.43	4343
CLAY	0.71	0.62	0.66	62908
SAND	0.74	0.77	0.75	109382
SILT	0.74	0.75	0.75	109199
accuracy			0.72	285832
macro avg	0.64	0.66	0.65	285832
weighted avg	0.73	0.72	0.72	285832

Figure 104. Accuracy report of resampled grain size dataset model

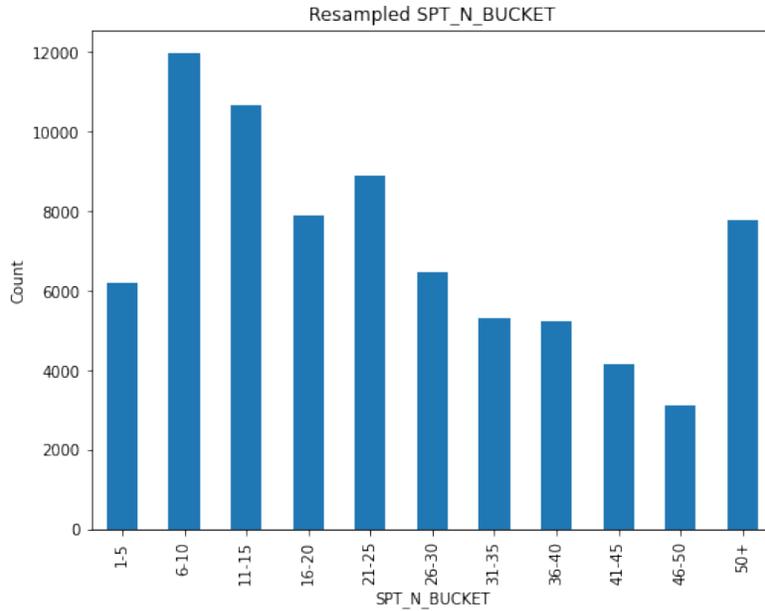


Figure 105. Histogram of the resampled SPTN dataset

	precision	recall	f1-score	support
1-5	0.60	0.54	0.57	3101
11-15	0.46	0.46	0.46	5325
16-20	0.45	0.33	0.38	3955
21-25	0.36	0.48	0.41	2968
26-30	0.44	0.35	0.39	2161
31-35	0.45	0.40	0.42	1767
36-40	0.38	0.48	0.42	1312
41-45	0.37	0.49	0.42	1038
46-50	0.33	0.50	0.40	781
50+	0.79	0.61	0.69	7786
6-10	0.49	0.62	0.55	5987
accuracy			0.51	36181
macro avg	0.46	0.48	0.47	36181
weighted avg	0.53	0.51	0.51	36181

Figure 106. Accuracy report of resampled SPTN model

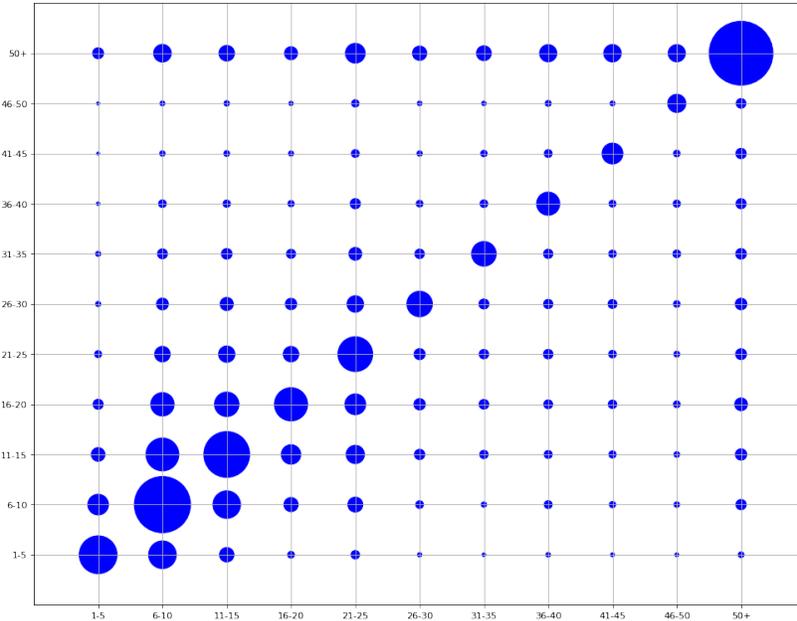


Figure 107. x-y scatter plot of resampled SPTN model

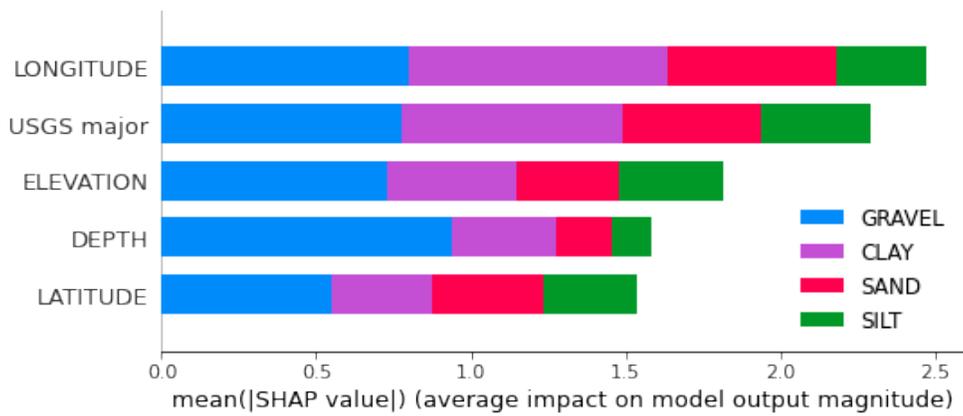


Figure 108. SHAP summary plot for MD grain size model

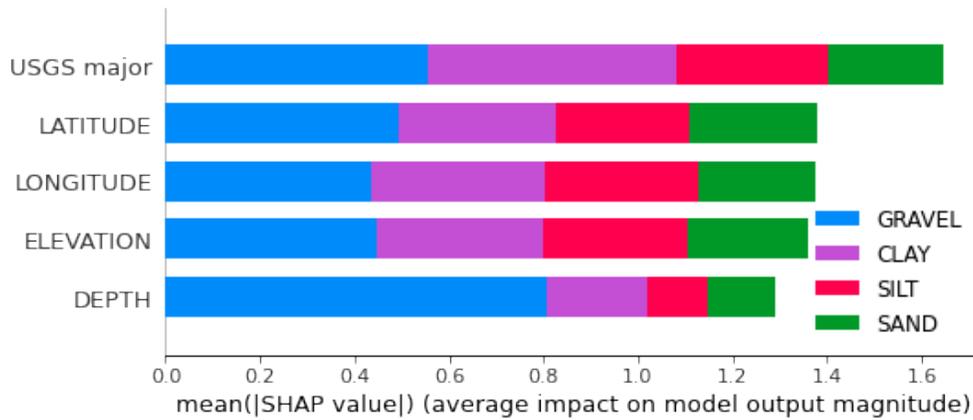


Figure 109. SHAP summary plot for consolidated grain size model

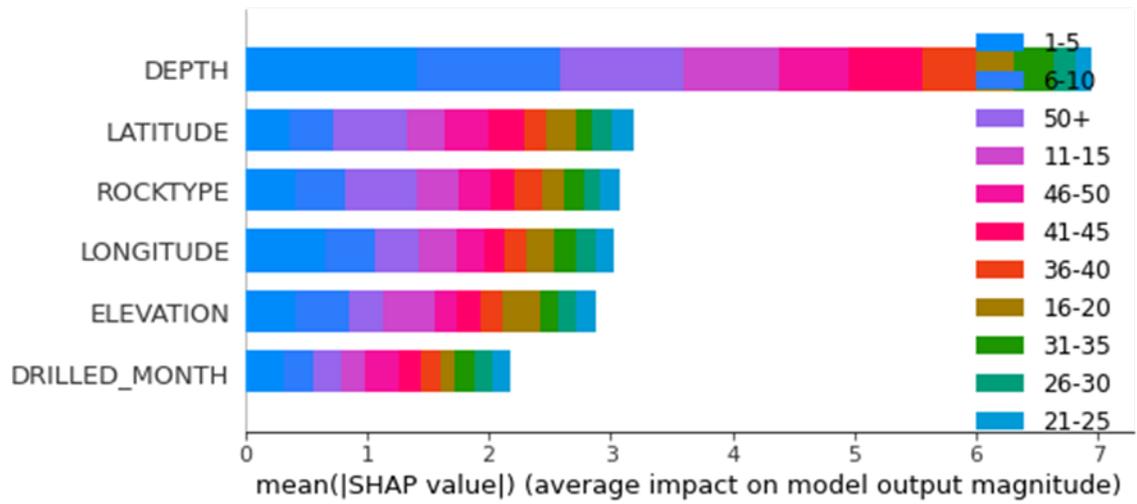


Figure 110. SHAP summary plot for SPTN model

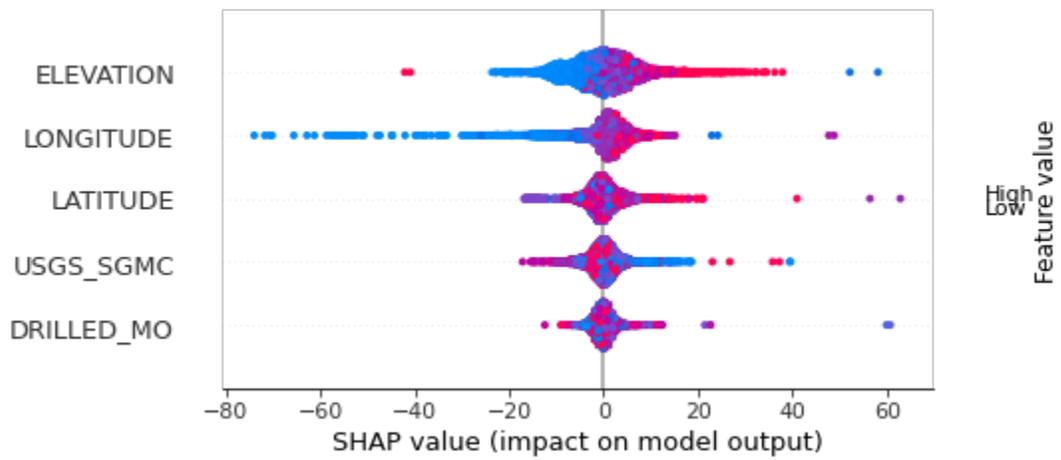


Figure 111. SHAP summary plot for ground water depth model

CHAPTER 8: SUMMARY & CONCLUSIONS

In this Phase II machine learning study, cutting-edge machine learning algorithms have been used to model the inherent complex relationship in selected highway datasets including LiDAR DEM data, concrete strength data from laboratory tests, drilling data from field tests, pavement marking retroreflectivity data collected by Laserlux mobile unit. Instance segmentation machine learning methods have been investigated and their prediction performance were tested for validity on automated generation of geotechnical inventory dataset including embankments and slopes as well as scarp distress lines. Team effort and coordination with MDOT SHA staff and engineers has been recognized as a very important factor to the success of this research, especially in identifying the research opportunities in highway datasets, retrieving and converting raw datasets from database servers to make them suitable for machine learning use, and providing professional guidance on machine learning model feature variables and desired output formats (e.g., concrete strength data and pavement marking retroreflectivity data). Three types of machine learning algorithms were studied, including neural network model for tabular data, LiDAR DEM derived slope raster images based random forest model for landslide risk assessment, instance segmentation models for embankment polygon and scarp distress lines detection applications. Specifically, the research team developed and tested machine learning models to make concrete compressive strength predictions by utilizing historical data from thousands of laboratory concrete strength tests done for various mix design and concrete placement variables. The research team also updated and tested machine learning model for several types of drilling data including SPT-N, grain size, and groundwater depth. A state-of-art instance segmentation machine learning model using ResNet algorithm were also tested for detecting objects of interest such as highway slopes and scarp distress lines in LiDAR DEM data derived raster tiff images for Maryland.

Machine learning models can be utilized by providing design engineers with historic data-based predictions. These predictions will be implemented to generate more accurate preliminary designs, which will result in significant fiscal savings. These machine learning methods will also help ease delays in material testing by giving data requesters information that can be immediately utilized.

It is imperative that we get these machine learning models in place to expedite the process of providing better preliminary data to MDOT SHA customers, thus improving customer service, fiscal savings, and reduced delays across all areas of OMT. Data size and quality is an important issue to ensure machine learning model performance and will thus be studied for the newly identified data as well. These models can be used to assist with the decision-making process of MDOT SHA in project planning and construction.

Based on the findings from this research, the suitability and applicability of each machine learning model type considered in this study of highway datasets are summarized below,

- A geospatial machine learning model has been applied to the statewide 2-m resolution LiDAR DEM data with the goal of mapping the landslide risks for Maryland. Trained with 256x256 cell-sized slope raster tiff chips based on historical slope failure data, the machine learning model was used to predict for landslide risk at a given location in Maryland and landslide risk maps for all counties in Maryland were generated. The map is presented in raster tiff formats and locations with risk score greater than 0.5 have been highlighted to inform engineers with increased landslide risk. Since a 2-m resolution LiDAR DEM data is used for generating the landslide risk map, landslide risk score can be found for each cell with a 2m x 2m small area in Maryland. This is considered high resolution landslide risk mapping in current practice. Preliminary inspection of the landslide risk map and comparing those areas with high landslide risk suggests that the machine learning predicted landslide risk results are consistent with the information from other sources such as USGS landslide risk map in general.
- Instance segmentation machine learning model was trained and successfully used to automatically identify and draw the embankment polygons in arc gis pro. Slope raster tiff images created from state-wide 2-m resolution LiDAR DEM data were considered appropriate for machine learning model enabled embankment and slope detection. Training data quality is critical to successful machine learning model predictions. Hyper-parameter tuning of machine learning model such as training image size and training epoch numbers also have significant effect on the optimal performance of embankment polygon detection machine learning model. It is noted that the total number of detected embankment polygons can be increased by lowering the confidence score threshold

value. However, it is general belief that those embankment polygons with higher confidence score value are more accurate. Post-processing of the identified embankment polygons such as filtering and merging are necessary. Validation of the machine learning identified embankment polygons were made by comparing with embankment inventory data provided by MDOT SHA.

- Instance segmentation machine learning model was also trained and applied to automatically detect scarp lines in arc gis pro. Surface parameter slope raster tiff images created from statewide 1-m resolution LiDAR DEM data were considered appropriate for machine learning model enabled scarp line detection. However, it is hypothesized that sub-meter resolution LiDAR DEM data will be even helpful in generating scarp lines using machine learning method (because scarp distress line typically has a very high slope value due to abrupt change in elevation at scarp surface) and future research should consider to use the higher resolution LiDAR DEM data such as 1-ft resolution. Training data quality is again found to be critical to successful machine learning model predictions. Hyper-parameter tuning of machine learning model such as training image size and other factors (e.g., 3-band 16-bits unsigned raster image format) also have significant effect on the optimal performance of scarp line detection machine learning model. Post-processing of the identified scarp line polygons such as filtering and centerline extraction are necessary. Validation of the machine learning identified scarp lines were made by comparing with scarp distress line inventory data provided by MDOT SHA.
- Tabular neural network machine learning model for concrete compressive strength predictions by utilizing historical data from thousands of laboratory concrete strength tests done for various mix design and concrete placement variables. Fast.ai neural network model was adopted in this study because Fast.ai model has demonstrated the state-of-art performance in the tabular data modeling in this study. Tabular data here refers to text (or ascii data) data typically stored in a SQL database and spreadsheet file. Two types of feedforward neural network models are available in Fast.ai for tabular data modeling: regression model and classification model. Both classification and regression machine learning models were developed for the concrete test data. For classification model, the training target is to classify the data samples into corresponding discrete

classes. The output of a regression model is nonetheless a continuous variable. Data cleanup was conducted to ensure that only the most reliable data was used in training a tabular neural network regression model. The concrete strength regression machine learning model prediction yields a R^2 value equal to 0.87 when tested on validation data. For the classification machine learning model to predict the confidence score of passing the 28-day concrete strength test, graphs of true value of concrete strength from laboratory test vs. confidence score of machine learning classification. The purpose of this classification machine learning model is that by looking at confidence score (e.g., below a certain threshold value), it is decided whether there are merits to do the actual strength test if the confidence score from machine learning model prediction is fairly low. This can be clearly seen in the plotted graph of true concrete strength value vs. machine learning prediction confidence score as most of the data are in either Quadrant #1 or Quadrant #3 of the graph while very small percentage of data points fall within Quadrant #2 for both Mix3 and Mix6 data. This distribution pattern suggests that if 0.5 confidence score is used as the dividing line between passing or not passing the strength test, those predictions with confidence score < 0.5 have a very high chance not to pass the strength.

- Laserlux collected pavement marking retroreflectivity data was also used to train machine learning regression models and predict the retroreflectivity values for various marking materials. Feature variables include county name, route number, age, materials and AADT for the regression machine learning model. Correct value of pavement marking installation date is critical to the accuracy of their age (or duration) value. Shapefile for the data frame was created to visualize the geospatial distribution of pavement marking data in Arc gis pro. The machine learning prediction results suggests that the general approach is very promising in predicting the pavement marking retroreflectivity values at given locations and time. Due to limited number of pavement marking retroreflectivity collected by Laserlux mobile unit (started two years ago), the regression machine learning model has an R^2 value of 0.7 when tested on real data. The feasibility of using machine learning model to establish the pavement marking condition deterioration model was also demonstrated. The machine learning derived deterioration curve is better than traditional statistical approach derived regression model as it provides

more details and can model complex retroreflectivity variation pattern over time (e.g., varying change rate over time).

- The research team trained tabular data neural network models for various geotechnical drilling datasets, including locations, geology, elevation, major soil constituent, layer depth, groundwater table depth, SPT N-Value ranges, and bedrock depth. The tabular data neural network model has been updated for the drilling data based on parametric study of hyper-parameters value as well as adopting an oversampling strategy to address the data imbalance issue in the original dataset. Tuning the machine learning model for optimal hyper-parameter values was conducted. Comparison of the predicted results from Fast.ai neural network model and true value was also made by calculating the corresponding confusion matrices and plotting a scatter plot in this study. SHAP analysis was also conducted to show the importance ranking of input features for the drill data machine learning models.

REFERENCES

- Bengio, Y. LeCun Y. and Hinton G. (2021). “Deep learning for AI” , Communications of the ACM, July 2021, 64(7): 58 - 65.
- Breiman, L. (2001). Random Forests. *Mach. Learn.* 45, 5–32.
- Maxwell, A.E.; Sharma, M.; Kite, J.S.; Donaldson, K.A.; Thompson, J.A.; Bell, M.L.; Maynard, S.M. (2020) “Slope Failure Prediction Using Random Forest Machine Learning and LiDAR in an Eroded Folded Mountain Belt.” *Remote Sensing* 12(3), 486.
- Youssef, A.M.; Pourghasemi, H.R.; Pourtaghi, Z.S.; Al-Katheeri, M.M. (2016). “Landslide susceptibility mapping using random forest, boosted regression tree, classification and regression tree and general linear models and comparison of their performance at Wadi Tayyah Basin, Asir Region, Saudi Arabia.” *Landslides*, 13, 839–856.
- Fast.ai (2022). Tabular model, <https://docs.fast.ai/tabular.model.html>
- Goodfellow, I., Bengio, Y. and Courville, A. (2016). *Deep Learning*. MIT Press, Boston, Massachusetts. ISBN: 9780262035613.
- Glorot, X., & Bengio, Y. (2010). “Understanding the difficulty of training deep feedforward neural networks.” *In Aistats* (Vol. 9, pp. 249-256).
- Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun. (2015). “Deep Residual Learning for Image Recognition,” <https://arxiv.org/abs/1512.03385>.
- Ioffe, S., C. Szegedy, (2015). “Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift,” <https://arxiv.org/abs/1502.03167>
- Lawal, M.O. (2021). “Tomato detection based on modified YOLOv3 framework,” *Scientific Reports*, vol. 11, Article number: 1447.
- LeCun, Y., Bengio, Y., & Hinton, G. (2015). “Deep learning.” *Nature*, 521(7553), 436-444.
- Liu, H. and Zhang, Y. (2020). “Bridge condition rating data modeling using deep learning algorithm,” *Structure and Infrastructure Engineering*, Published online, <https://doi.org/10.1080/15732479.2020.1712610>
- Liu, H. and Zhang, Y. (2019). “Image-driven structural steel damage condition assessment method using deep learning algorithm.” *Measurement*, 133: 168-181.
- Mao, H., Alizadeh, M., Menache, I, and Kandula, S. (2016). “Resource Management with Deep Reinforcement Learning,” *HotNets-XV*, November 9-10, 2016, Atlanta, GA, USA.
- Redmon, J. and A. Farhadi. (2018). “YOLOv3: An Incremental Improvement,” *arXiv:1804.02767*.

Yang, Y., Kaiwen Zha, Chen, Y.-C., Wang, H., & Katabi, D. (2021). Delving into Deep Imbalanced Regression. ArXiv. Web link: <https://doi.org/10.48550/arxiv.2102.09554>

Lundberg, S. M., & Lee, S. (2017). “A unified approach to interpreting model predictions.” Neural Information Processing Systems, Vol. 30, 4768–4777.

Iwright. (2022). Understanding machine learning with SHAP analysis. Acerta.ai. Retrieved April 27, 2023, web link: <https://acerta.ai/blog/understanding-machine-learning-with-shap-analysis/>

Smith, J., Jones, R., & Lee, K. (2020). “The Influence of Ambient Temperature on High Performance Concrete Properties.” Journal of Materials Science and Engineering, 8(3), 32-39.