17th World Conference on Earthquake Engineering, 17WCEE
Sendai, Japan - September 13th to 18th 2020

ON THE USE OF MACHINE LEARNING TECHNIQUES TO PREDICT LATERAL SPREADING DISPLACEMENT IN NEW ZEALAND

E. Rathje(1), M. G. Durante(2)

(1) *Professor, Department of Civil, Architectural, and Environmental Engineering, University of Texas at Austin, USA, e.rathje@mail.utexas.edu*
(2) *Postdoctoral Fellow, Department of Civil, Architectural, and Environmental Engineering, University of Texas at Austin, USA, mgdurante@utexas.edu*

Abstract

In the last decade the availability of a large amount of high-quality data from post-disaster field reconnaissance generated substantial datasets of structural, infrastructural, and geotechnical damage. Such large datasets enable artificial intelligence approaches that can provide insights into the physical behavior of soil, and it can be instrumental for improving liquefaction hazard analysis. The 2010-2011 Canterbury earthquake sequence in New Zealand caused significant structural damage and widespread liquefaction over a large region. Following this earthquake sequence, remote sensing techniques were used to produce lateral spreading displacement maps [1]. Such data, combined with the availability of site characterization data from the New Zealand Geotechnical Database (NZGD), constitutes a unique resource for lateral spreading hazard analysis. This large liquefaction-induced lateral spreading dataset includes more than 1700 data points. Each data point is characterized by one cone penetration test (CPT) profile and co-located lateral spread displacement information. Building upon this substantial dataset, we develop machine learning-based empirical models to predict lateral spread occurrences. We first identify and distill key physics-based model input parameters. We then test and compare various machine learning algorithms trying to minimize the discrepancies between model prediction and observed occurrence. Such computationally expensive analyses were carried out utilizing cloud-based computing capabilities offered by DesignSafe [2]. Our preliminary analysis shows that machine learning techniques can be successfully utilized to predict lateral spread occurrences in New Zealand. The evaluation of the amount of displacement and the scalability and applicability of such models for global applications will be investigated in future research efforts.

Keywords: machine learning; liquefaction; lateral spreading; 2010-2011 Canterbury earthquake sequence.

1. Introduction

Soil liquefaction is a phenomenon that typically occurs in saturated loose sandy soils subjected to rapid loading conditions, such as earthquakes. The generation of excess pore water pressure is a direct consequence of the rapid loading, that can lead to a sudden reduction in the strength and stiffness of the soil. In the presence of gently sloping ground or near the free face of a slope, the earthquake-induced liquefaction generates lateral displacement, known as lateral spreading. The horizontal displacement induced by lateral spreading typically causes significant damage to the built environment.

The 2010-2011 Canterbury earthquake sequence in New Zealand caused significant structural damage and widespread liquefaction over a large region, with extensive lateral spreading observation in the area adjacent to the Avon River [3]. Following this earthquake sequence, remote sensing techniques were used to produce lateral spreading displacement maps [1]. Such maps are obtained through image correlation of satellite imagery acquired before and after the earthquake. The availability of site characterization data in the area of interest from the New Zealand Geotechnical Database (NZGD) combined with these maps represents a unique resource for lateral spreading hazard analysis. Such a substantial dataset enables artificial intelligence approaches. In the dataset, each lateral spread horizontal displacement data point is associated with one cone penetration test (CPT) profile. The goal of this research is to develop machine learning-based empirical models to predict liquefaction-induced lateral spreading displacement. The approach can be divided into two parts: a model that predicts the occurrence of lateral spreading (or lack thereof), and a second model that predicts the amount of displacement.

This paper focuses on the first part of the model (i.e., predicting the occurrence of lateral spreading) using data from the 22 February 2011 Christchurch earthquake (M_w 6.2). Figure 1 shows lateral spreading horizontal displacements generated by this event in the area of the Avon River, eastern Christchurch from [1].

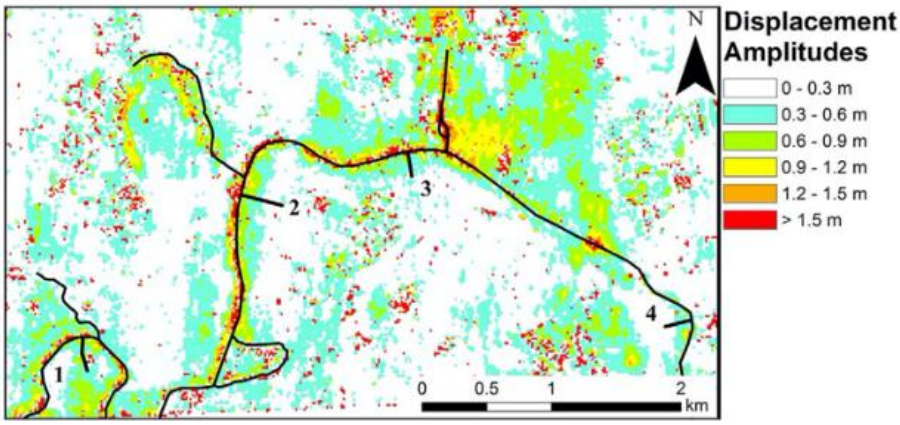

Fig. 1 – Lateral spreading horizontal displacement for the Christchurch earthquake computed from Light Detection and Ranging (LiDAR) surveys (after Rathje et al. 2017).

In this study we use information relative to 1700 data points in the Avon River area to develop models that predict the occurrence of lateral spreading. The type of information collected are geometrical (i.e. distance from the river, ground slope), event-specific conditions (i.e. Peak Ground Acceleration (PGA), Ground Water Table - GWT) and subsurface soil conditions (i.e., CPT profiles). The size of the dataset enabled the use of Machine Learning (ML) techniques to predict the occurrence of lateral spreading. Two models of increasing complexity were developed in this study: (1) a model that does not use CPT data and (2) a complete model that includes CPT data. Additionally, we compare the performance of two different ML algorithms: a Random Forest, tree-based approach and a kernel-based Support Vector Machine approach. All analyses were performed in the cloud on DesignSafe [2]. Such models were developed in a Jupyter Notebook [4] using python and the, NumPy, Pandas, Matplotlib, and Scikit-learn packages [5-8].

2. Dataset

Figure 2 shows the general study area and the geographical distribution of various input parameters used in the study to develop ML-based prediction models. Ground motion and displacement data are from the 22 February 2011 Christchurch earthquake (M_w 6.2). An extensive dataset of 1702 data points is considered in the area of interest. Each datapoint has information about location (longitude and latitude), distance of the point from the free face (L), depth of the ground water table (GWT) (Fig. 2a), ground slope calculated from a 5m LIDAR grid (Fig. 2b), PGA, and some representative quantities extracted from the CPT profiles available from the NZGD. The event specific GWT and PGA were extracted from the NZGD [9-10]. Information about the PGA are not used in this study due to the small variation of PGA in the area of interest, that could generate misleading results if used in a machine learning algorithm. At each location, a layer of 4m below the GWT is extracted from the CPT profile to compute the normalized tip resistance (Q_{tn}) and the soil behavior type index (I_c) profiles [11]. Median (Med) and standard deviation (σ) values, shown using heat map, are then calculated for each data point and included in the dataset (Fig. 2c-d).

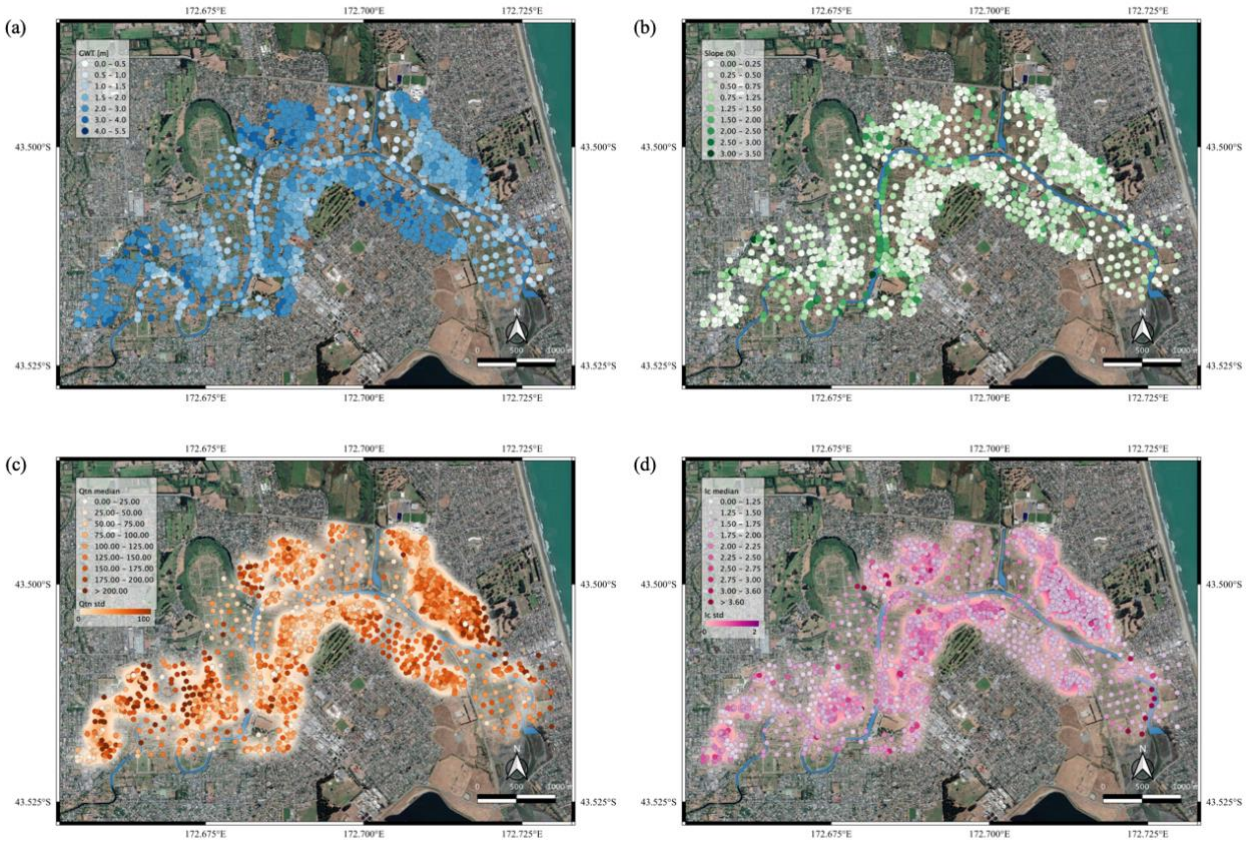

Fig. 2 – Data distribution: (a) GWT, (b) slope, (c) median normalized tip resistance (Q_{tn}) with standard deviation, and (d) median soil behavior type index (I_c) with standard deviation.

Lateral displacement occurrence is evaluated for each datapoint based on the remote sensing analyses presented by Rathje et al. 2017 [1] (Fig. 1). The threshold used to discern between the occurrence and non-occurrence of lateral spreading is set to 0.3m (i.e. if the lateral spread displacement is < 0.3m, we consider the datapoint as "no lateral spread", if the displacement is > 0.3m, we assign "yes lateral spread"). The datapoints are well distributed between the two classes, with 846 and 856 points in Class 0 (no lateral spread) and Class 1 (yes lateral spread), respectively. Figure 3 shows the distribution of lateral spreading occurrences for the dataset considered. The displacement classes used in Fig. 3b were selected to have as much equally distributed intervals as possible.

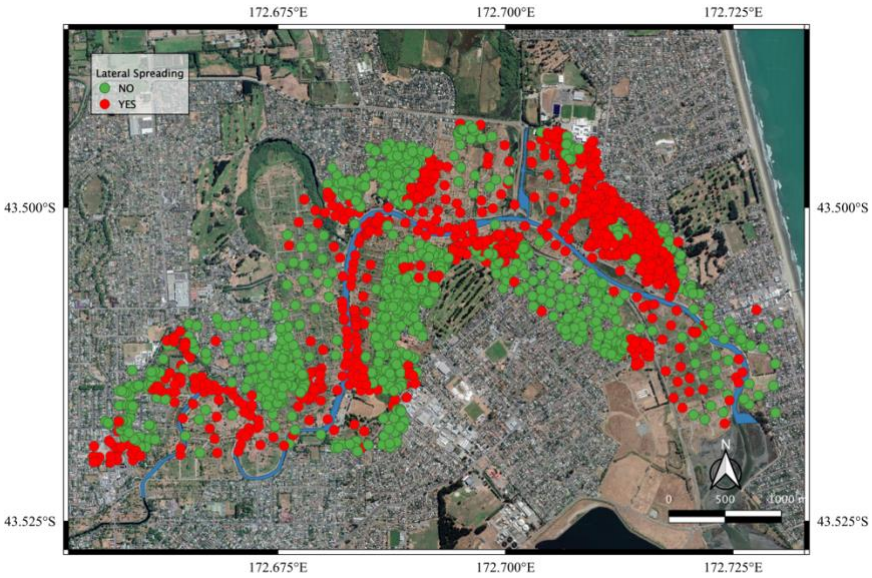

Fig. 3 – Data distribution for lateral spreading occurrence.

3. Machine Learning Techniques

Artificial Intelligence (AI) is a relatively new discipline [12] that studies how to train computers to execute specific tasks. Depending on the human intervention needed, AI algorithms can be classified as Machine Learning (ML) and Deep Learning (DL). A ML algorithm needs the data to be organized in a dataset with labelled data and it uses that data to make a prediction. A DL algorithm also needs labelled data, but it can learn on its own through various layers, hierarchies, and concepts (i.e., artificial neural networks) suitable to perform complex calculations, and, for this reason, it requires significantly more data. This paper focuses only on ML methods. It is possible to identify three types of ML algorithms: *(i)* supervised - task driven, in which the outcome or output for the given input is known; *(ii)* unsupervised - data driven, in which the outcome or output for the given inputs is unknown; and *(iii)* reinforcement, in which the model learns from mistakes. Supervised learning models are used for classification or regression problems. In classification problems, the model will predict the categorical response (class). Such models can be binary or multi-class. In regression problems, the model will predict a continuous response value. Supervised machine learning models are used in this study to solve a classification problem, i.e., lateral spreading Yes/No. In an AI analysis, each variable considered is called a feature, while the prediction is called a target. The features used in this problem are all the entries of the dataset described in the previous section, and the target is the occurrence (or not) of lateral spreading.

The two algorithms presented in this paper are tree-based and kernel-based methods (Fig. 4). The simplest tree-based method is a decision tree, which involves the division of the prediction space into smaller regions using simple yes/no questions, each capable of capturing different relations in the dataset. A decision tree can have multiple layers of questions (depth) to predict the response. The trade-off with decision trees is that too few layers may result in an inaccurate prediction, while too many layers may be result in overfitting. The Random Forest (RF) approach [8] addresses this issue by developing multiple decision using different parts of the datasets and/or a subset of features. The combination of all the trees generated by the model defines the final prediction model, running each row through each tree, collecting the value at the end node (also called leaf node), and taking the response with more votes for a classification model. RF algorithms can also be used to define the importance of each feature based on its predictive power. Kernel-based models analyze the data to find specific patterns in the input dataset. The kernel-based algorithm used in this study is the Support Vector Machine model (SVM) [8]. Such methods transform the space in a map and the data into vectors. The relationship between inputs and outputs is represented by a user-specified kernel, also called a similarity function.

To guarantee a proper validation of a ML model, the dataset is usually subdivided into training and test data. The division can be done before running the algorithm, or it can be done multiple times during the analysis using Cross Validation (CV) methods. The basic approach, called k-fold CV [8], is an automated procedure that divides the dataset randomly into k smaller sets or folds. For each random division, the method uses (k-1) folds to train the model, and the last fold to validate it. This procedure is repeated k times and the performance of the model is computed as the average of each of them. In this paper the k parameter for the CV is set to 10. For each ML model, a hyperparameter optimization algorithm is used to set the optimal set of model parameters [8]. In the RF models the hyperparameters considered are: *(i)* maximum depth of each tree, *(ii)* number of estimators (trees), *(iii)* maximum number of features considered for node splitting, and *(iv)* the function to measure the quality of a dataset split (criterion). In the SVM models the hyperparameters considered are: *(i)* the kernel function, *(ii)* the kernel coefficient (gamma), *(iii)* the function regularization parameter (C).

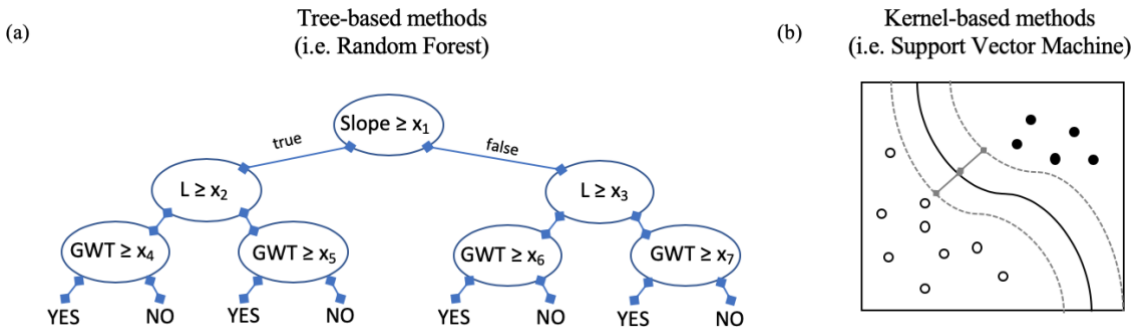

Fig. 4 –Schematic representation of (a) tree-based and (b) kernel-based methods.

The quality of a ML model can be evaluated from the corresponding Confusion Matrix (CM). A CM is a matrix representation of the results that is used to visualize the performance of the algorithm in each class (or values). Each row represents the instances in a class and each column represents the instances in the predicted class. For the specific binary case of Yes/No lateral spreading, if the model prediction is Yes and the observed class is also Yes, the point is classified as a True Positive (TP). If the prediction is No and the observation is also No, the point is classified as a True Negative (TN). If the prediction is Yes, but the observation is No, it is a False Positive (FP). Finally, if the prediction is No, but the observation is Yes, it is classified as a False Negative (FN). The better the model performs, the fewer points are found in the FP and FN segments of the confusion matrix. Additional information on the overall performance of a model can be obtained looking at some specific metrics such as: *(i)* accuracy, *(ii)* recall, *(iii)* precision, and *(iv)* F1-measure. Accuracy looks at the overall performance, assuming equal costs for both kinds of errors (Eq. 1). Recall focuses on the capability of the model to correctly identify the class (Eq. 2). Precision indicates how good the model is at generating only a small number of FP (Eq. 3). The F1-measure is a metric able to combine recall and precision (Eq.4).

$$\text{Accuracy} = (TP + TN) / (TP + TN + FP + FN) \quad (1)$$

$$\text{Recall} = TP / (TP + FN) \quad (2)$$

$$\text{Precision} = TP / (TP + FP) \quad (3)$$

$$\text{F1-measure} = (2 \times \text{Recall} \times \text{Precision}) / (\text{Recall} + \text{Precision}) \quad (4)$$

4. Results

This section discusses results obtained applying ML algorithms to the New Zealand liquefaction-induced lateral spreading dataset presented in the previous sections. Results are divided into two sections: the first part uses only a portion of the features available in the dataset and compares the performance of the RF and SVM

algorithms. The features not considered in the first part of the analysis are those obtained from CPT data. The second section presents results obtained from the RF algorithm that uses all the features collected in the dataset, including CPT data. Results are presented by means of normalized confusion matrixes, table of metrics, and maps reporting the spatial distribution of TP, TN, FP, and FN among the data points.

4.1 Random Forest versus Support Vector Machine algorithms (no CPT data)

The tree-based RF algorithm and the kernel-based SVM algorithm are used to predict the occurrence of liquefaction-induced lateral spreading in the Avon River area in eastern Christchurch using the features distance from the free face (L), ground slope (S), and depth to the GWT (GWT). The term Support Vector Classification (SVC) is used to identify the SVM used in classification models. The selection of the best parameters for each algorithm uses the hyperparameter optimization algorithm based on the model accuracy. Figure 5 shows the normalized confusion matrixes for the best model found for the (a) RF and (b) SVC algorithms. The CMs show that RF performs better in all the sections of the matrix. The total accuracy for the RF model is 88%, while the accuracy of the SVC model is 65%. Additional information about the overall performance of the two models are presented in Table 1 in terms of recall, precision, and F1-score. These metrics are all around 88% for the RF model, and they range from 61% to 69% for the SVC model.

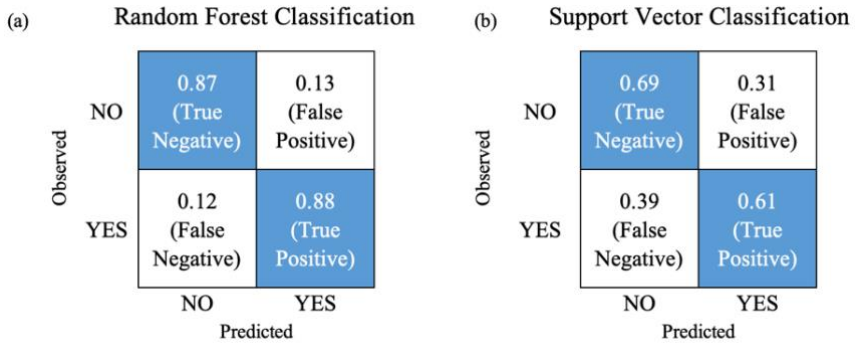

Fig. 5 – Normalized confusion matrixes for (a) Random Forest Classification and (b) Support Vector Classification algorithms.

Table 1 – Main metrics for RF and SVC algorithms

	Accuracy		**Recall**		**Precision**		**F1-score**	
	RF	SVC	RF	SVC	RF	SVC	RF	SVC
Class 0	-	-	0.87	0.69	0.88	0.63	0.87	0.66
Class 1			0.88	0.61	0.87	0.66	0.88	0.63
average	0.88	0.65	0.88	0.65	0.88	0.65	0.88	0.65

Figure 6 shows the distribution of the RF and SVC model predictions in terms of TP (green dots), TN (green triangles), FP (red dots), and FN (yellow triangles) among the data points. Errors in the predictions are not concentrated in one specific zone of the study area, meaning that both models are able to recognize general patterns in the data. The RF model (Fig. 6a) performs significantly better than SVC (Fig. 6b) mainly in the eastern part of the domain, where there are large regions of FN and FP predictions in the SVC model. The better performance of the RF model is attributed to the capability of this algorithm to find hidden relationships among features using deeper decision trees.

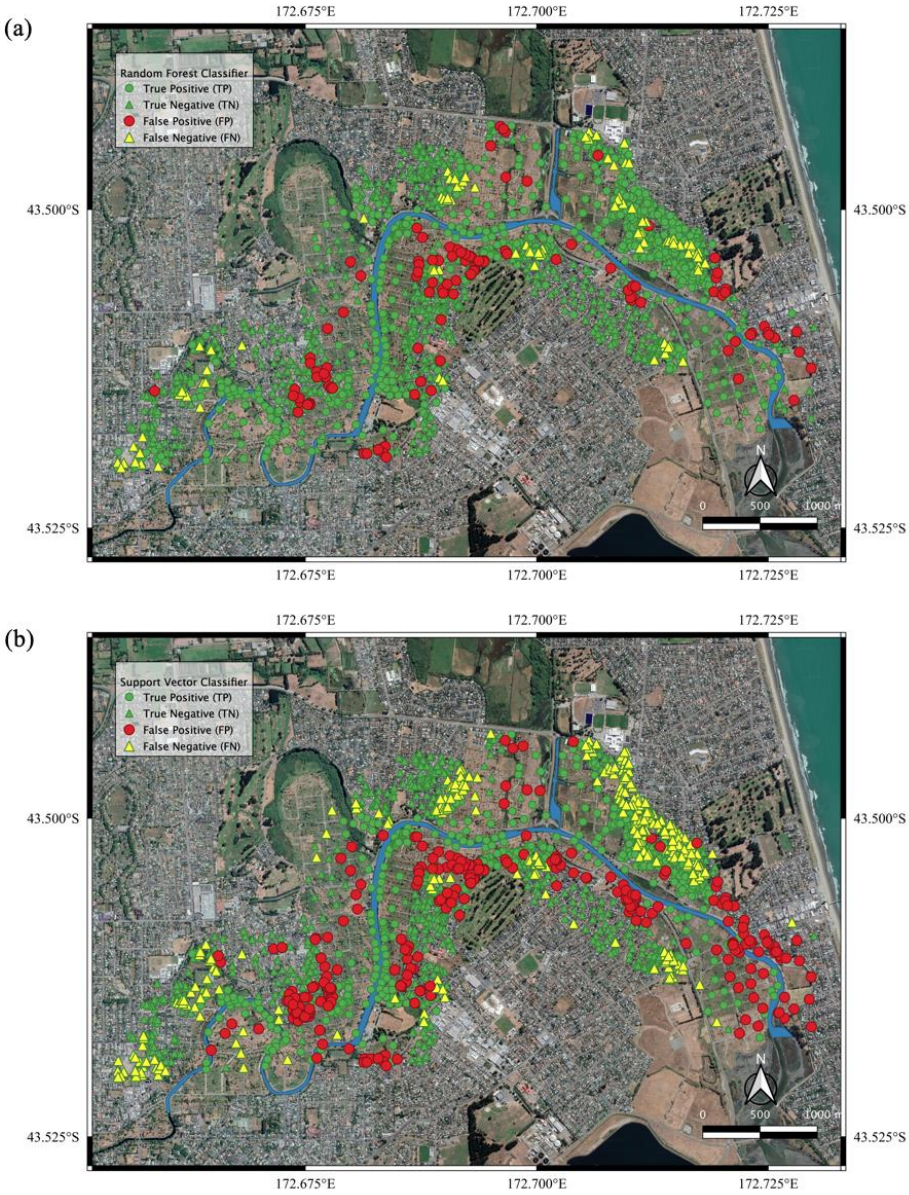

Fig. 6 – Data distribution for (a) Random Forest and (b) Support Vector Classification algorithms.

4.2 Random Forest algorithm with CPT data

In this section a RF algorithm is used to predict the occurrence of lateral spreading based on all the features available in the dataset. The dataset includes Q_{tn} and I_c (Meds and σs) extracted for the 4-meter layer below the GWT. Figure 7 compares the CM of the RF model without the CPT data with the RF model obtained considering the CPT data in the analysis. The comparison shows that the inclusion of some information about strength and soil type (through Q_{tn} and I_c, respectively) improves the overall performance of the predictive model. As shown in Fig. 7, the inclusion of CPT data increases the TN and TP responses, and reduces the FN and FP responses. The most improvement is observed for TP and FN. The accuracy of the model marginally increases from 88% to 91%. Additional information about the overall performance of the two models are presented in Table 2. The 3% increase of the performance of the model after the CPT data are considered, is distributed among all the metrics, as reported in Table 2. It is important to include CPT data as they provide

information on the mechanical properties of the soil. However, in this case, the improvement of the model is not significant, due to the small variation in the mechanical properties of the soil within the analyzed region. In fact, in a ML model, the addition of features with small variation does not introduce enough information into the algorithm to improve significantly its overall performance.

Figure 8 compares the relative importance of each of the features obtained from the RF models with and without the CPT data. Figure 8a shows that GWT and L are the two most important features for the model without CPT data with 39% and 35% of importance, respectively. These two features are still the top ranked features for the RF model with CPT data (Fig. 8b), with a relative importance of 24% and 20%, respectively. For the dataset considered, the other features have similar importance values between 9% and 13%, with I_c (Med) as the most important feature derived from CPT data with 13% relative importance.

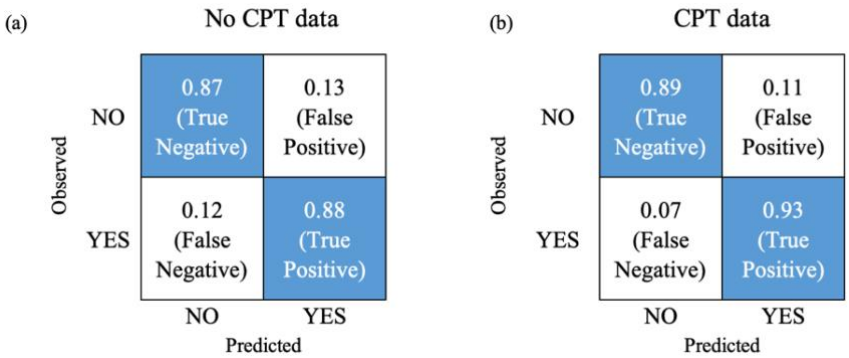

Fig. 7 – Normalized confusion matrixes from Random Forest Classification (a) without CPT data and (b) with CPT data.

Table 2 – Main metrics for RF algorithms with and without CPT data

	Accuracy		**Recall**		**Precision**		**F1-score**	
	No CPT	CPT	No CPT	CPT	No CPT	CPT	No CPT	CPT
Class 0	-	-	0.87	0.89	0.88	0.93	0.87	0.91
Class 1			0.88	0.93	0.87	0.90	0.88	0.91
average	0.88	0.91	0.88	0.91	0.88	0.91	0.88	0.91

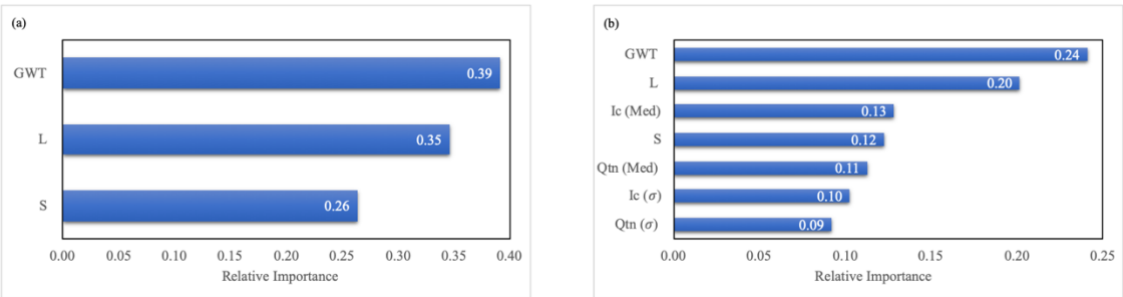

Fig. 8 – Relative feature importance ranking from Random Forest Classification (a) without CPT data and (b) with CPT data.

Figure 9 compares the distribution of FP (red dots), and FN (yellow triangles) among the datapoints for the RF models with and without the CPT data. The RF model with CPT data (Fig. 9b) reduces the overall error in the prediction. The comparison between Fig. 9a and 9b shows that the area with the CPT data improved the

performance especially in the eastern area, where the majority of the FP were concentrated when no CPT data were used in the model.

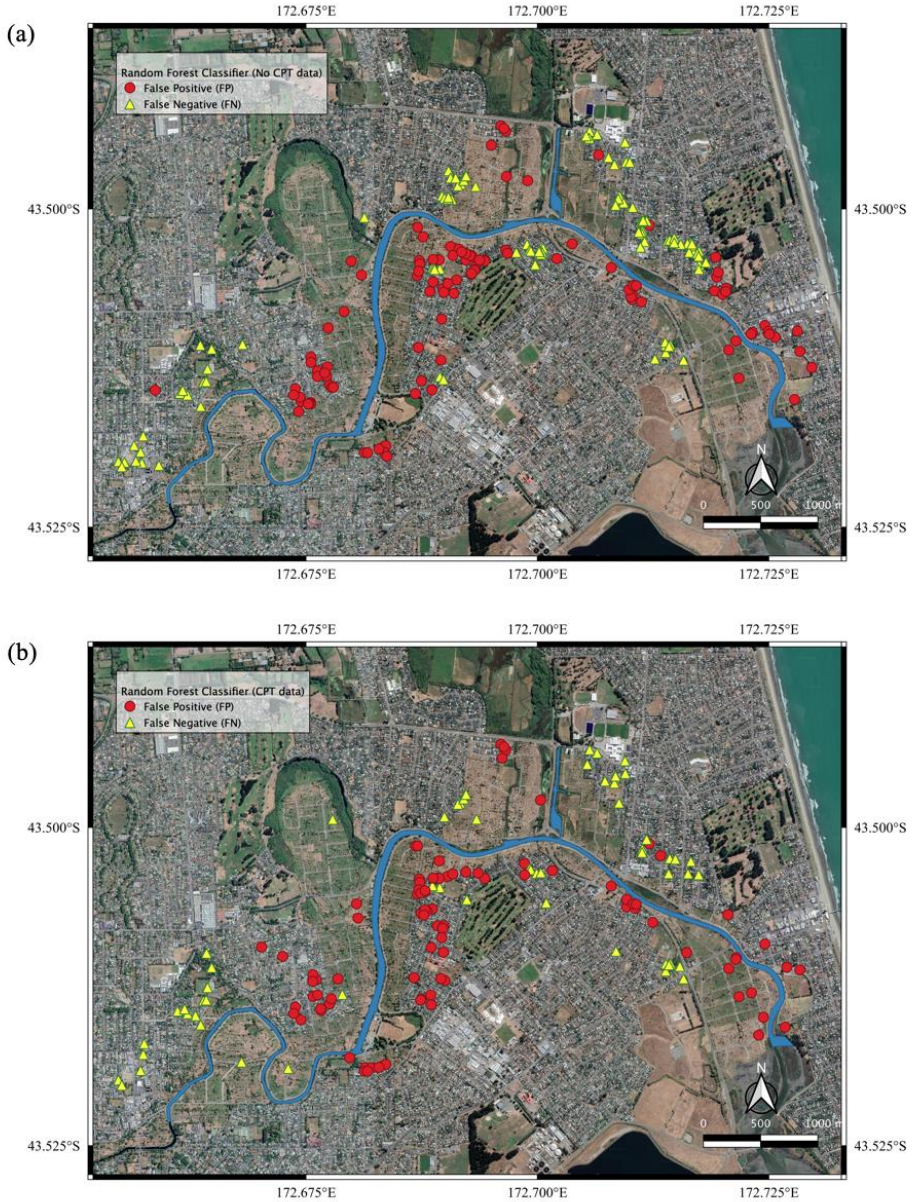

Fig. 9 – False Positive (FP) and False Negative (FN) Random Forest model results for dataset (a) without CPT data and (b) with CPT data.

5. Conclusions

The use of machine learning techniques to predict the occurrence of liquefaction-induced lateral spreading displacement in New Zealand is investigated in this paper. Data from the 22 February 2011 Christchurch earthquake (Mw 6.2) are used to create a dataset with more than 1700 data points in the Avon River area. Machine Learning algorithms are used to predict the lateral spread displacement occurrences, dividing the dataset into two classes: Class 0 where no lateral spread displacement was observed (displacement < 0.3m), and Class 1 otherwise. The ML algorithms used in this study are the tree-based Random Forest (RF) and the kernel-based Support Vector Classification (SVC) algorithms. All analyses were performed in the cloud on

DesignSafe, in Jupyter Notebooks. The comparison of the overall performance of the RF and SVC algorithms is performed using a reduced feature set that does not include the information obtained from the CPT profiles. The analyses show that for this dataset the RF model better predicts the occurrence of lateral spreading. The inclusion of the CPT data in the dataset improved the performance of the model by 3%. The small improvement of the model observed in this case study may be attributed to the small variation in the soil properties within the area of interest, which does not provide enough additional information to the model. This study shows that machine learning techniques can be successfully used to predict lateral spread occurrences. This paper is part of a more comprehensive on-going project. Future studies will focus on the prediction of the amount of displacement and the scalability and applicability of such models in different regions.

6. References

[1] Rathje, E. M., Secara, S. S., Martin, J. G., van Ballegooey, S., and Russel, J. (2017): Liquefaction-Induced Horizontal Displacements from the Canterbury Earthquake Sequence in New Zealand Measured from Remote Sensing Techniques. *Earthquake Spectra*, **33** (4), 1475-1494. https://doi.org/10.1193/080816EQS127M.

[2] Rathje, E., Dawson, C. Padgett, J.E., Pinelli, J.-P., Stanzione, D., Adair, A., Arduino, P., Brandenberg, S.J., Cockerill, T., Dey, C., Esteva, M., Haan, Jr., F.L., Hanlon, M., Kareem, A., Lowes, L., Mock, S., and Mosqueda, G. (2017): DesignSafe: A New Cyberinfrastructure for Natural Hazards Engineering. *ASCE Natural Hazards Review*, doi:10.1061/(ASCE)NH.1527-6996.0000246.

[3] Cubrinovski, M., Robinson, K., Taylor, M., Hughes, M., and Orense, R. (2012): Lateral spreading and its impacts in urban areas in the 2010-2011 Christchurch earthquakes. *New Zealand J. of Geology and Geophysics*, **55** (3), 255–269, DOI:10.1080/00288306.2012.699895.

[4] Perez F., Granger B.E. (2007): IPython: a system for interactive scientific computing, *Computing in Science & Engineering* **9**, 21–29.

[5] Oliphant, T. E. (2006): A guide to NumPy (Vol. 1). *Trelgol Publishing USA*.

[6] McKinney, W., & others. (2010): Data structures for statistical computing in python. In *Proceedings of the 9th Python in Science Conference*, **445**, 51–56.

[7] Hunter, J. D. (2007): Matplotlib: A 2D graphics environment. *Computing in Science & Engineering*, **9** (3), 90–95.

[8] Pedregosa et al. (2011): Scikit-learn: Machine Learning in Python, *JMLR* **12**, 2825-2830.

[9] New Zealand Geotechnical Database (2014) "Event Specific Groundwater Surface Elevations", Map Layer CGD0800 – 12 June 2014, retrieved [date] from https://www.nzgd.org.nz/

[10] New Zealand Geotechnical Database (2015) "Conditional PGA for Liquefaction Assessment", Map Layer CGD5110 – 30 June 2015, retrieved [date] from https://www.nzgd.org.nz/

[11] Robertson, P. K., and Wride, C. E. (1998): Evaluating cyclic liquefaction potential using the cone penetration test. *Can. Geotech. J.*, **35** (3), 442–459.

[12] McCarthy, John, M.L. Minsky, N. Rochester, C.E. Shannon (1955): A proposal for the Dartmouth summer conference on artificial intelligence, *Conference Announcement*.

