Performance Analysis of Machine Learning Algorithms in Storm Surge Prediction

Vai-Kei Ian[1][a], Rita Tse[1,2], Su-Kit Tang[1,2][b] and Giovanni Pau[1,3,4][c]
[1]*Faculty of Applied Sciences, Macao Polytechnic University, R. de Luís Gonzaga Gomes, Macao SAR, China*
[2]*Engineering Research Centre of Applied Technology on Machine Translation and Artificial Intelligence of Ministry of Education, Macao Polytechnic University, R. de Luís Gonzaga Gomes, Macao SAR, China*
[3]*Department of Computer Science and Engineering - DISI, University of Bologna, Via Zamboni, 33, 40126 Bologna, Italy*
[4]*UCLA Computer Science Department, 404 Westwood Plaza, Los Angeles, CA, U.S.A.*

Keywords: Storm Surge, Machine Learning, Ensemble Machine Learning Algorithm, Natural Disaster.

Abstract: Storm surge has recently emerged as a major concern. In case it occurs, we suffer from the damages it creates. To predict its occurrence, machine learning technology can be considered. It can help ease the damages created by storm surge, by predicting its occurrence, if a good dataset is provided. There are a number of machine learning algorithms giving promising results in the prediction, but using different dataset. Thus, it is hard to benchmark them. The goal of this paper is to examine the performance of machine learning algorithms, either single or ensemble, in predicting storm surge. Simulation result showed that ensemble algorithms can efficiently provide optimal and satisfactory result. The accuracy of prediction reaches a level, which is better than that of single machine learning algorithms.

1 INTRODUCTION

The use of machine learning (ML) involves algorithms and statistical models that can learn through inference and pattern recognition (Selvam and Babu, 2015), and that can adapt, without being explicitly programmed, to the changing environments. Recent advances in machine learning have led to its application to many domains, solving various types of problems (Chan et al., 2021b; Chan et al., 2021a; Lin et al., 2021; Tse et al., 2020; Cheok et al., 2022).

Ensemble ML algorithms use multiple bases of ML algorithms to give a common estimate of a result in order to reduce the generalization error. In theory, the prediction error should be decreased if the base prediction models are sufficiently independent and diverse. Stability and accuracy are then increased by minimizing error caused by factors such as noise, bias, and volatility (Lessmann et al., 2015). In other words, it augments and improves overall prediction accuracy over that of single algorithms, by combining results from numerous models. The combined result is usually better in terms of prediction accuracy when compared to utilizing a single ML algorithm. Therefore, ensemble ML algorithms are also widely used in a variety of applications and fields.

Prediction of storm surge in risk assessment has been one of the difficult problems for years due to the complex structure of storm surge and variety of influencing factors being developed while it is progressing. Applying machine learning (ML) techniques in risk assessment continuously evolves due to their ability to capture the associated relationships efficiently. Thus, various machine learning algorithms have been proposed to predict storm surges previously. However, quality of prediction is hardly compared, as datasets of different structure and data type were used.

In this paper, single and ensemble ML algorithms will be evaluated using a storm surge dataset for prediction of possible storm surge occurrence in the South China Sea region. The dataset, collected from Hong Kong (HKO, 2021) and Kaohsiung (CWB, 2021) on storm scenarios from 2017 to 2020, will be used for training the algorithm models with hyperparameters tuned using 10-fold cross-validation. Evaluation result on testing showed that ensemble ML algorithms have advantages over single ML algorithms, achieving a satisfactory level of certainty and confi-

[a] https://orcid.org/0000-0003-2505-1173
[b] https://orcid.org/0000-0001-8104-7887
[c] https://orcid.org/0000-0003-2216-7170

Ian, V., Tse, R., Tang, S. and Pau, G.
Performance Analysis of Machine Learning Algorithms in Storm Surge Prediction.
DOI: 10.5220/0011109400003194
In *Proceedings of the 7th International Conference on Internet of Things, Big Data and Security (IoTBDS 2022)*, pages 297-303
ISBN: 978-989-758-564-7; ISSN: 2184-4976

dence.

The remainder of the paper is organized as follows. In section 2, a background of ensemble ML algorithms in prediction will be introduced. After that, we give the evaluation configuration and result in section 3. In section 4, a discussion on the limitations of the simulation will be given. Finally, the conclusion of the work is provided in section 5.

2 RELATED WORK

The application of machine learning (ML) algorithms in risk assessment for storm surges are continuously evolving due to their ability to capture the associated relationships efficiently. Various machine learning algorithms have been proposed to predict for storm surges previously (Sarzaeim et al., 2017; Wu et al., 2019; Kim et al., 2019; Modaresi et al., 2018; Ni et al., 2020; Sankaranarayanan et al., 2020; de Oliveira and de Carvalho Carneiro, 2021; Theera-Umpon et al., 2008; Yu et al., 2006). To improve prediction accuracy, ensemble ML algorithms have been proposed.

Max Voting (Arafat et al., 2019), Averaging, and Weighted Average (Shahhosseini et al., 2022) are typical basic ensemble techniques, whereas Bagging, Boosting, Blending, and Stacking (Dou et al., 2020; Franch et al., 2020) are prominent advanced ensemble approaches. These strategies can improve and enhance performance when more patterns are observed and the final prediction is a consensus from the models that comprise it. For example, Bagging, or bootstrap aggregations, is the process of creating models in parallel, which might be similar or different, and averaging their related predictions as the final result. Boosting, on the other hand, refers to the sequential construction of models by repeatedly evaluating the success of ancestral models in a hierarchy. The subordinate level of model emphasizes the learning process for estimate in cases when previous models failed to perform successfully. AdaBoost, Gradient Boosting, and XGBoost are three popular approaches (Mahesh, 2020). While Stacking seeks to link models by including their outputs as features into the final model, Blending is quite similar in that it leverages those base models to deliver predictions as new features, with the final model being trained on the new features to yield the ultimate prediction. The sole distinction is that in Blending, the meta model is trained on a separate holdout set rather than a complete and folded training set. In other words, Stacking employs out-of-fold predictions for the next layer's training dataset, whereas Blending uses a validation set instead for the subsequent layer's training.

Both single ML and ensemble ML algorithms can give promising and useful result in storm surge prediction. However, their performance in prediction is not fairly comparable as different datasets were used in their training and testing. In this paper, using one dataset with various attributes from tropical cyclones, tide levels, and meteorological conditions, for training and testing, the performance of single ML and ensemble ML algorithms are evaluated.

3 PERFORMANCE EVALUATION

To evaluate the performance of ML algorithms (single or ensemble), a high-quality dataset with appropriate attributes is crucial. It will be injected into both single and ensemble ML algorithms individually for training, testing, and validation of performance in terms of accuracy. Single ML algorithms include Decision tree, Naive Bayes, K-nearest neighbors, XGBoost and SVM; whereas ensemble ML algorithms include Bagging, Random Forest, AdaBoost, Gradient boost, Voting and Stacking. Configuration and comparative results will be discussed further in section 3.2.

3.1 Dataset

In this paper, the storm surge dataset (HKO, 2021; CWB, 2021) is used in evaluation as it contains various important attributes from tropical cyclones, tide levels, and meteorological conditions. In total, there are 17 columns of features that are used for training and validating models. Among them, grade, central pressure, maximum wind, distance, azimuth angle, mean sea level pressure (MSLP), changes in mean sea level pressure, wind speed, changes in wind speed, and abnormal surge level are included as they are the most significant attributes in determining the possibility of the occurrences of storm surge. A detailed description is shown below.

- Grade: Categories of tropical cyclones, including 1. Low (L), 2. Tropical Depression (TD), 3. Tropical Storm (TS), 4. Severe Tropical Storm (STS), 5. Typhoon (TY)
- Central Pressure: Surface pressure at the center of the tropical cyclone as estimated or measured
- Max Wind: Maximum value of the average wind speed at the surface
- Distance: Distance apart calculated by the latitude and longitude of the tropical cyclone

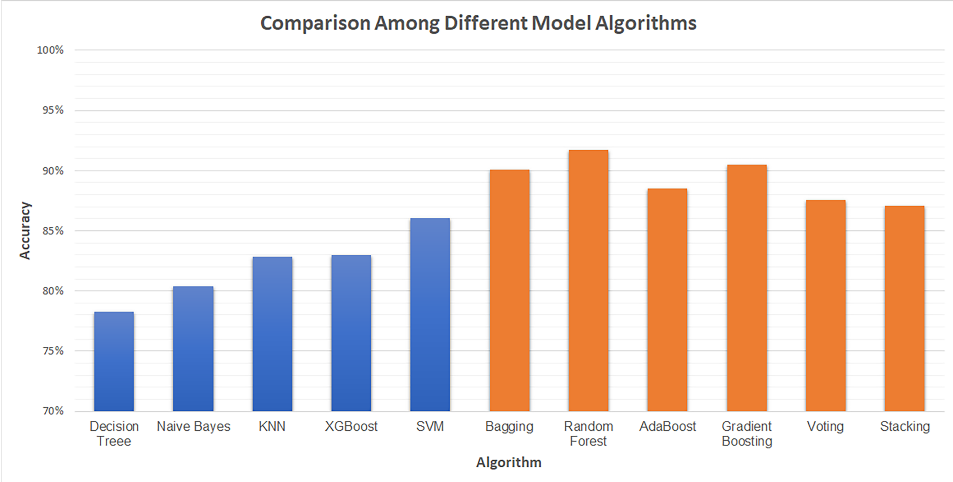


Figure 1: Performance comparison on accuracy among different ML algorithms.

- Azimuth Angle: The azimuth angle calculated by the direction towards which the center of the typhoon is moving
- MSLP/WS: Measurement of the mean sea level pressure/wind speed at ground station
- MSLP/WS Delta: Changes in mean sea level pressure/wind speeds measurements over the past hours
- Surge: Difference between the actual water level and the normal sea water level (astronomical tide)

As the amount of storm surge data in the dataset is limited, it is divided into several parts for *k*-fold cross-validation (CV), which is an efficient approach in data utilization during model building. The accuracy is ensured and noise influence is minimized across the development cycle (Lee et al., 2020; Jung, 2018). For generalization of accuracy, it allows unbiased estimates by validating a prediction model with unseen data.

For ensemble ML algorithms, the input dataset is divided into *k* samples. One of the *k* samples will be designated as the test set, while the remaining *k - 1* samples will be designated as the training set. Every sample is subjected to a recursive cycle on all classifiers. The performance of the algorithms could then be summarized across all *k* trials collectively, demonstrating the accuracy of prediction on unseen data.

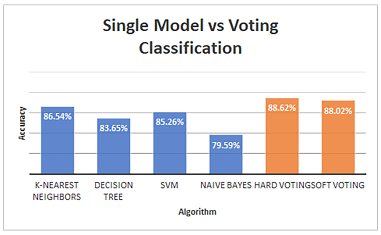
3.2 Configuration

In the evaluation, an intensive simulation was carried out to compare the performance of single and ensemble ML algorithms. The simulation environment was setup with a high-end computer running CentOS Linux with Python and open-sourced ML libraries (Géron, 2019; Stančin and Jović, 2019; Raschka et al., 2020). The hardware and software configuration for the simulation environment is listed below.

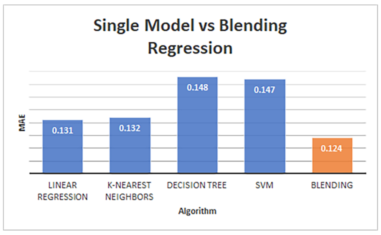
1. Processing (CPU): 3.20GHz Intel® Core™ i7-8700 (4.60GHz boost) 6 cores, 12 threads, with 12MB Intel® Smart Cache
2. Graphics (GPU): NVIDIA Quadro P1000 (4GB GDDR5)
3. RAM: 32GB (2667MHz)
4. Storage: 450GB SSD + 930GB HDD
5. OS: CentOS Linux
6. Language: Python 3.9
7. Libraries: Numpy, Scipy, Pandas, Scikit-learn, TensorFlow, Keras, Matplotlib

3.3 Results

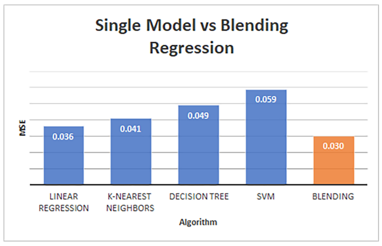
Using the storm surge dataset introduced in section 3.1, experiments on the 5 single ML algorithms and 6 ensemble ML algorithms introduced in section 3 with data re-sampling using *k*-fold CV were conducted and summarized in Figure 1. The results revealed that ensemble ML algorithms could expectantly give optimal prediction outcomes against the single ML algorithms. In addition, some of the algorithms have been selected for performance evaluation by using Mean Absolute Error (MAE) and Mean Squared Error (MSE) (Gaudette and Japkowicz, 2009; Kumar et al., 2020) along with accuracy obtained respectively. Results complied with the overall trend for performance behaviors where satisfactory results could be obtained from ensemble algorithms, as can be seen in Figure 2a, 2b & 2c. However, it does not apply to all problem-solving strategies. In Figure 2d, 2e & 2f, some single ML algorithms show similar performance



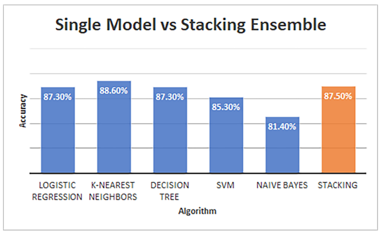
(a) Better performance in Ensembles.



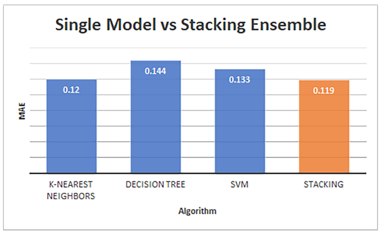
(b) Finer results in MAE.



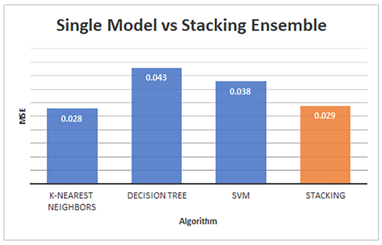
(c) Finer results in MSE.



(d) Similar performance.



(e) Similar results in MAE.



(f) Similar results in MSE.

Figure 2: Performance level acquired from single/ensemble ML methods.

with ensemble algorithms in terms of accuracy, MAE & MSE.

3.4 Adjustments to Ensemble Methods

For machine learning algorithms, choosing an appropriate level of model complexity is a key balancing act. If a model is too complex, data will be fit into the model entirely, resulting in poor generalization to unseen data (overfitting) in testing. If its complexity is too low, it would not capture all information in the data (underfitting). To improve the stability and accuracy of the prediction model by reducing errors caused by variables such as noise, bias, and variability when ensemble algorithms are adopted becomes critical and essential. To tackle this issue, hyperparameter tuning can be applied, which can create a significant impact on behaviors and accuracy of the model, by adjusting its associated parameters for the best outcome.

Ensemble algorithms have gained a lot of attention and are extensively utilized because they can outperform a single ML approach in general. These approaches used the notion of re-sampled or re-weighted training data sets from the original data, and a prediction model was applied to each of them many times. Boosting, for example, has been designed to increase the performance of any weak learning algorithm by dynamically modifying the distribution of the training data set and taking a weighted majority vote on their predictions. The bagging approach, on the other hand, uses bootstrap samples to produce classifiers in an ensemble. Each of them is based on random sampling with the same number of instances as the original data. The ultimate result is reached by majority voting. Meanwhile, random forest was developed as an ensemble approach for combining tree classifiers in such a way that each level of trees is determined by the values of their features, which are selected from a set of features sampled individually. When dividing a node, the chosen split is no longer the best split across all features during tree building. Instead, the split is determined by selecting the best option from a random subset of the characteristics. Because of this unpredictability, the bias generally increases significantly, but the variance reduces owing to the averaging effect.

To demonstrate, the ideal size of decision trees should be viewed in terms of the overall performance of ensemble techniques during which decision trees are used as model classifiers. The number of target variable categories that can explain the nature of the input dataset is the most common tree size in boosting. It demonstrates that the error rate of a stump tree, the number of classes, and the depth of a single tree may be used to determine the ideal tree size for a given dataset. In general, greater tree sizes produce more accurate findings. In other words, the accuracy of ensemble trees increase as tree size increases.

To evaluate the important affects on accuracy of several ensemble techniques by tree size, its effects on two types of ensemble methods are distinct. First, the tree size impact is not uniform in the boosting type ensemble technique. The ideal tree size might be modest depending on the dataset. In contrast to boosting, bootstrap-based ensemble approaches (bagging

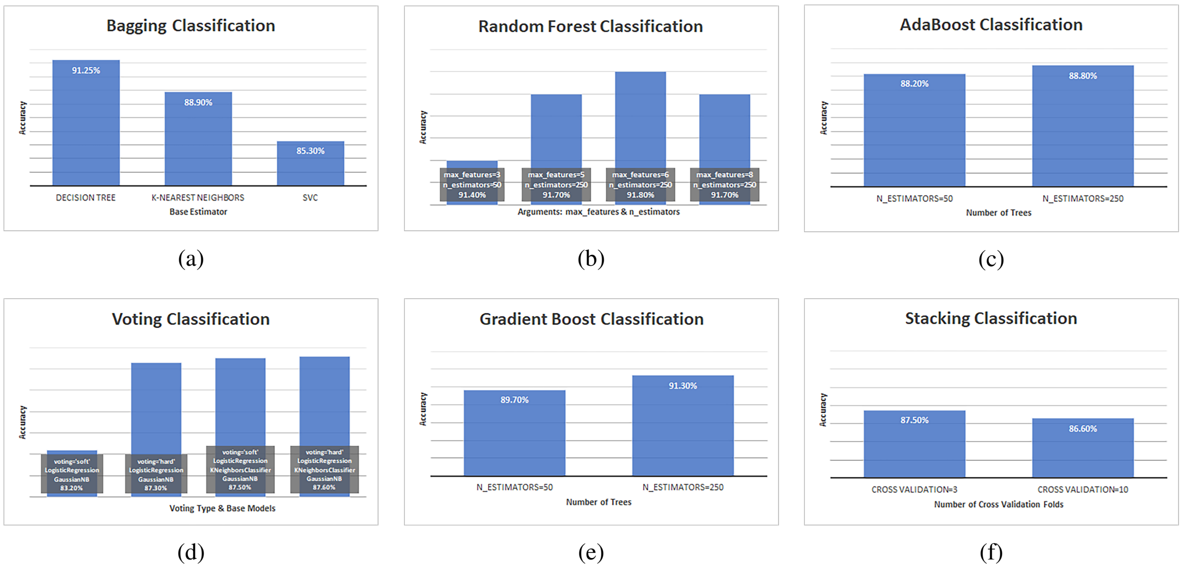


Figure 3: Parameter adjustments directly affect performance for different ensemble ML algorithms.

and random forest) produce more uniform outcomes. In most scenarios, a greater number of tree sizes results in higher accuracies. Changes to the depth of a tree, or the size and number of trees in the forest, as shown in Figure 3, would improve the overall performance. It was shown than by having a proper adjustment and optimal configuration in base estimator, all ensemble models would have a satisfactory level of accuracy, with Random Forest having the best performance model at 91.80%. On the other hand, stacking appears to be the least efficient in this comparative experiment, with an accuracy of just 87.5%, even though its flexibility and adaptability ease the development of its prediction model.

4 DISCUSSION AND LIMITATIONS

Principal benefits to using ML approach in storm surge prediction would be its excellent temporal efficiency. The proposed ensemble ML algorithms are intuitive to use and very time effective, which is a significant advantage over other current classical prediction systems. It is not necessary to prepare any documentation in advance for prediction events. As a result, interpretation of the prediction findings is mostly attainable without the need for expert assistance. When provided accessible typhoon forecast information, which could be collected automatically from the Internet, each prediction could be made in 1–2 minutes, including gathering typhoon information, data pre-processing, calculation, and visualization, using a widely available PC. In comparison, if the storm surge forecast was processed in the traditional method, it would take significantly longer. The ensemble ML approach's performance exhibits its excellent temporal efficiency. Users may also adjust the predicted location and maximum wind speed in terms of tropical cyclone intensity forecast uncertainty due to the speedy processing time. Overall, the suggested ML algorithms, which are effective and efficient in their operational application, respond quickly to the need for emergency consultation and might give timely auxiliary decision-making support.

ML algorithms also enable the concept of multi-scenario prediction. When the anticipated course of the typhoon position deviates much from the actual movement pattern, benefits of ensemble ML prediction models are highlighted as the models can incorporate all conceivable typhoon positions, taking into account not only the uncertainty of forward heading direction but also variance in speed of movement. Thus, a generalized and objective prediction result on the possibility for having storm surges could be derived.

Aside from the essential relevance and effectiveness of ensemble ML algorithms in storm surge predictions, precision in various other elements such as astronomical tide computation and tropical cyclone forecasts form a critical basis for the success of the ensemble ML approaches. The variability of the weather system is the major source of prediction error. Specifically, the route and severity of typhoons might abruptly shift, making precise surge elevation problematic. As a result, the first and most important goal is to unravel the mechanism of these fluctuations and include them accordingly into the prediction model in order to improve forecast accuracy and enhance performance. In general, such abnormal tracks may be traced to the underlying surface condition, interaction with other weather systems, and the presence of related atmospheric circulation in the absence of environmental guiding flow. When a typhoon passes over a warm body of water, strength changes are directly connected with air-sea interaction. When the sea surface temperature rises, the mixing layer thickens, or the interaction time lengthens, typhoons build quickly because more heat can be absorbed. The movement of a typhoon, for example, would be meandering. As a result, because the locations and intensities of the typhoons are crucial features in the overall building of the prediction model, this driving factor affects the overall performance of the model. Second, considering the long-term implications of climate change is essential to the appropriate operation of the prediction model. The effects of climate change on storm surges are multidimensional. On the one hand, the frequency and severity of Typhoons have increased in recent decades. During a storm, a rise in wind intensity might result in more severe tragedies. On the other hand, sea level is progressively increasing as a result of receding glaciers and the melting ice in the Arctic. Storm surge impacts and consequences would be boosted proportionately. As a result, the success of the prediction model should be able to reflect and capture these changes, as well as the retraining of the model with newly related collected data, in order to produce and reflect a suitable assessment for the possibility of this natural catastrophe occurring.

Future research should consider the potential effects, characteristics and significant parameters of storm surge more carefully, especially under the influences of global climate change and sea level rise. Effects of future climate change could be addressed by collection of more atmospheric data from IoT devices, such as smart lampposts, spread across the territories which enable the prediction models to capture recent changes in our climate and provides more reliable prediction results. This is one of the key components in future attempts to mitigate storm surge hazards. To get the most out of these additional efforts, combine a more effective and efficient sampling strategy during model retraining with sufficient, adequate and balanced training dataset. We believe that the proposed ensemble prediction methods could further be extended and adjusted for specific coastal applications, such as providing immediate operational surge forecasts, probabilistic coastal flood hazard assessments, or future surge forecasts during typhoon seasons. Future studies should also address and investigate the shifting patterns of storm surges over time, as well as the impacts of the wind intensity field, based on the findings of this paper.

5 CONCLUSION

As storm surge datasets vary considerably among studies, benchmarking machine learning algorithms, both single and ensemble, using the same dataset can reveal their performance in terms of prediction accuracy. This paper compares the performance of the algorithms and highlights the significance of ensemble machine learning algorithms in storm surge prediction. In our simulation, we found that the ensemble machine learning algorithms, Random forest classification and Stacking, performs better than single and other ensemble machine learning algorithms in storm surge prediction. In case of the occurrence of overfitting and underfitting in training, its prediction result will contain bias, which can be resolved by hyperparameter tuning to the algorithm.

ACKNOWLEDGEMENTS

This work was supported by the Macao Polytechnic University – Edge Sensing and Computing: Enabling Human-centric (Sustainable) Smart Cities (RP/ESCA-01/2020).

REFERENCES

Arafat, M. Y., Hoque, S., Xu, S., and Farid, D. M. (2019). Machine learning for mining imbalanced data. *IAENG International Journal of Computer Science*, 46(2):332–348.

Chan, K. I., Chan, N. S., Tang, S.-K., and Tse, R. (2021a). Applying gamification in portuguese learning. In *2021 9th International Conference on Information and Education Technology (ICIET)*, pages 178–185. IEEE.

Chan, N. S., Chan, K. I., Tse, R., Tang, S.-K., and Pau, G. (2021b). Respect: privacy respecting thermal-based specific person recognition. In *Thirteenth International Conference on Digital Image Processing (ICDIP 2021)*, volume 11878, page 1187802. International Society for Optics and Photonics.

Cheok, S. M., Hoi, L. M., Tang, S.-K., and Tse, R. (2022). Crawling parallel data for bilingual corpus using hybrid crawling architecture. *Procedia Computer Science*, 198:122–127.

CWB (2021). CWB Observation Data inquire System. https://www.cwb.gov.tw/eng/.

de Oliveira, L. A. B. and de Carvalho Carneiro, C. (2021). Synthetic geochemical well logs generation using ensemble machine learning techniques for the brazilian pre-salt reservoirs. *Journal of Petroleum Science and Engineering*, 196:108080.

Dou, J., Yunus, A. P., Bui, D. T., Merghadi, A., Sahana, M., Zhu, Z., Chen, C.-W., Han, Z., and Pham, B. T. (2020). Improved landslide assessment using support vector machine with bagging, boosting, and stacking ensemble machine learning framework in a mountainous watershed, japan. *Landslides*, 17(3):641–658.

Franch, G., Nerini, D., Pendesini, M., Coviello, L., Jurman, G., and Furlanello, C. (2020). Precipitation nowcasting with orographic enhanced stacked generalization: Improving deep learning predictions on extreme events. *Atmosphere*, 11(3):267.

Gaudette, L. and Japkowicz, N. (2009). Evaluation methods for ordinal classification. In *Canadian conference on artificial intelligence*, pages 207–210. Springer.

Géron, A. (2019). *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems.* " O'Reilly Media, Inc.".

HKO (2021). HKO Open Data. https://www.hko.gov.hk/en/abouthko/opendata_intro.htm.

Jung, Y. (2018). Multiple predicting k-fold cross-validation for model selection. *Journal of Nonparametric Statistics*, 30(1):197–215.

Kim, S., Seo, Y., Rezaie-Balf, M., Kisi, O., Ghorbani, M. A., and Singh, V. P. (2019). Evaluation of daily solar radiation flux using soft computing approaches based on different meteorological information: peninsula vs continent. *Theoretical and Applied Climatology*, 137(1):693–712.

Kumar, R., Kumar, P., and Kumar, Y. (2020). Time series data prediction using iot and machine learning technique. *Procedia computer science*, 167:373–381.

Lee, J.-Y., Choi, C., Kang, D., Kim, B. S., and Kim, T.-W. (2020). Estimating design floods at ungauged watersheds in south korea using machine learning models. *Water*, 12(11):3022.

Lessmann, S., Baesens, B., Seow, H.-V., and Thomas, L. C. (2015). Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research. *European Journal of Operational Research*, 247(1):124–136.

Lin, H., Tse, R., Tang, S.-K., Chen, Y., Ke, W., and Pau, G. (2021). Near-realtime face mask wearing recognition based on deep learning. In *2021 IEEE 18th Annual Consumer Communications & Networking Conference (CCNC)*, pages 1–7. IEEE.

Mahesh, B. (2020). Machine learning algorithms-a review. *International Journal of Science and Research (IJSR).[Internet]*, 9:381–386.

Modaresi, F., Araghinejad, S., and Ebrahimi, K. (2018). A comparative assessment of artificial neural network, generalized regression neural network, least-square support vector regression, and k-nearest neighbor regression for monthly streamflow forecasting in linear and nonlinear conditions. *Water Resources Management*, 32(1):243–258.

Ni, L., Wang, D., Wu, J., Wang, Y., Tao, Y., Zhang, J., and Liu, J. (2020). Streamflow forecasting using extreme gradient boosting model coupled with gaussian mixture model. *Journal of Hydrology*, 586:124901.

Raschka, S., Patterson, J., and Nolet, C. (2020). Machine learning in python: Main developments and technology trends in data science, machine learning, and artificial intelligence. *Information*, 11(4):193.

Sankaranarayanan, S., Prabhakar, M., Satish, S., Jain, P., Ramprasad, A., and Krishnan, A. (2020). Flood prediction based on weather parameters using deep learning. *Journal of Water and Climate Change*, 11(4):1766–1783.

Sarzaeim, P., Bozorg-Haddad, O., Bozorgi, A., and Loáiciga, H. A. (2017). Runoff projection under climate change conditions with data-mining methods. *Journal of Irrigation and Drainage Engineering*, 143(8):04017026.

Selvam, V. and Babu, R. (2015). An overview of machine learning and its applications. *International Journal of Electrical Sciences & Engineering (IJESE)*, 1(1):22–24.

Shahhosseini, M., Hu, G., and Pham, H. (2022). Optimizing ensemble weights and hyperparameters of machine learning models for regression problems. *Machine Learning with Applications*, page 100251.

Stančin, I. and Jović, A. (2019). An overview and comparison of free python libraries for data mining and big data analysis. In *2019 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, pages 977–982. IEEE.

Theera-Umpon, N., Auephanwiriyakul, S., Suteepohnwiroj, S., Pahasha, J., and Wantanajittikul, K. (2008). River basin flood prediction using support vector machines. In *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, pages 3039–3043. IEEE.

Tse, R., Mirri, S., Tang, S.-K., Pau, G., and Salomoni, P. (2020). Building an italian-chinese parallel corpus for machine translation from the web. In *Proceedings of the 6th EAI International Conference on Smart Objects and Technologies for Social Good*, pages 265–268.

Wu, L., Peng, Y., Fan, J., and Wang, Y. (2019). Machine learning models for the estimation of monthly mean daily reference evapotranspiration based on cross-station and synthetic data. *Hydrology Research*, 50(6):1730–1750.

Yu, P.-S., Chen, S.-T., and Chang, I.-F. (2006). Support vector regression for real-time flood stage forecasting. *Journal of hydrology*, 328(3-4):704–716.