Relative Importance of Radar Variables for Nowcasting Heavy Rainfall: A Machine Learning Approach

Yi Victor Wang, Seung Hee Kim, *Member, IEEE*, Geunsu Lyu, Choeng-Lyong Lee, Gyuwon Lee, Ki-Hong Min, and Menas C. Kafatos, *Life Member, IEEE*

***Abstract*— Highly short-term forecasting, or nowcasting, of heavy rainfall due to rapidly evolving mesoscale convective systems (MCSs) is particularly challenging for traditional numerical weather prediction (NWP) models. To overcome such a challenge, a growing number of studies have shown significant advantages of using machine learning (ML) modeling techniques with remote sensing data, especially weather radar data, for high-resolution rainfall nowcasting. To improve ML model performance, it is essential first and foremost to quantify the importance of radar variables and identify pertinent predictors of rainfall that can also be associated with domain knowledge. In this study, a set of MCS types consisting of convective cell (CC), mesoscale CC, diagonal squall line (SLD), and parallel squall line (SLP), was adopted to categorize MCS storm cells, following the fuzzy logic algorithm for storm tracking (FAST), over the Korean Peninsula. The relationships between rain rates and over 15 variables derived from data products of dual-polarimetric weather radar were investigated and quantified via five ML regression methods and a permutation importance algorithm. As an applicational example, ML classification models were also developed to predict locations of storm cells. Recalibrated ML regression models with identified pertinent predictors were coupled with the ML classification models to provide early warnings of heavy rainfall. Results imply that future work needs to consider MCS type information to improve ML modeling for nowcasting and early warning of heavy rainfall.**

***Index Terms*— Artificial neural network (ANN), convolutional neural network (CNN), deep learning, dual-polarimetric weather radar, early warning, flash flood, hydrometeorological hazard, Lasso, mesoscale convective system (MCS), permutation importance, random forest, remote sensing, storm, support vector regression (SVR).**

Manuscript received 9 August 2022; revised 29 October 2022 and 8 December 2022; accepted 15 December 2022. Date of publication 20 December 2022; date of current version 10 January 2023. This work was supported by the Korea Meteorological Administration under Grant KMI2020-00910. *(Corresponding author: Seung Hee Kim.)*

Yi Victor Wang, Seung Hee Kim, and Menas C. Kafatos are with the Institute for Earth, Computing, Human and Observing, Chapman University, Orange, CA 92866 USA (e-mail: ywang2@chapman.edu; sekim@chapman.edu; kafatos@chapman.edu).

Geunsu Lyu and Choeng-Lyong Lee are with the Center for Atmospheric Remote Sensing (CARE), Kyungpook National University, Daegu 41566, South Korea (e-mail: geunsulyu@gmail.com; lchly747@naver.com).

Gyuwon Lee and Ki-Hong Min are with the Department of Astronomy and Atmospheric Sciences, Kyungpook National University, Daegu 41566, South Korea (e-mail: gyuwon@knu.ac.kr; kmin@knu.ac.kr).

This article has supplementary downloadable material available at https://doi.org/10.1109/TGRS.2022.3231125, provided by the authors.

Digital Object Identifier 10.1109/TGRS.2022.3231125

I. Introduction

NOWCASTING is a highly short-term weather forecasting of up to a few hours [1]. It can provide useful information for short-term early warnings regarding hazardous meteorological conditions to support preparation for and rapid response to emergencies, as well as to mitigate the potential impacts, associated with a climate-related hazard event. For example, flash flooding is a common hazard that can cause severe adverse impacts to human communities [2], [3], [4]. Such phenomena usually occur within hours of mesoscale rainstorm events. An effective nowcasting system for mesoscale heavy rainfall can be used to issue warnings to reduce potential flood exposure of communities that may be caught off guard otherwise.

For nowcasting heavy rainfall of a mesoscale convective system (MCS), the numerical weather prediction (NWP) models that are traditionally used for daily weather forecasting have their limitations [5]. NWPs produce simulations of meteorological conditions based on physical equations coupled with measurements of atmosphere. When adopted for applications with high spatial and temporal resolutions, traditional NWP models are usually hampered by the deficiencies in determination of initial and boundary conditions [6], long spin-up time [7], and challenges in physical parameterization [8]. As a result, it is difficult for traditional NWP models to provide useful nowcasting products on MCS rainfall for high-resolution applications such as hydrometeorological modeling and simulation for an urban area or a catchment.

As an option to improve nowcasting of rainfall, machine learning (ML) methods can be applied along with remote sensing data. As remote sensing data such as from weather radars and satellites can usually reach a spatial and temporal resolution as high as 1 km and 5 min, the nowcasting results can be produced with the same level of detail. To fully integrate the detailed information from remote sensors, ML models can be trained on empirical data to interpolate and extrapolate rainfall given new measured values of input variables. A growing number of ML methods, including convolutional neural network (CNN) [9], [10], [11], [12], deep generative modeling [5], logistic regression [13], long short-term memory network [9], [10], [11], multilayer perceptron (MLP) network [14], random forest classification [15], [16], recurrent neural network [17],

support vector machine classification [18], trajectory gated recurrent unit network [19], and U-net [20], have been demonstrated with potential to accurately nowcast rainfall. Given a set of identified input variables, an ML model functions as a surrogate model that can swiftly predict or classify rainfall. The ML models based on convolutional computations are particularly good at capturing spatial correlations [9], [10], [11], [12]. The recurrent network-based ML methods are especially capable of modeling the temporal trends [9], [10], [11]. Despite the proved utility of ML models in nowcasting rainfall, the input variables of the current models are usually selected without empirical quantification of the importance of variables based on evidence from data before model training.

Among the recent efforts on ML modeling for nowcasting rainfall, weather radars are the most commonly used data sources for input variables. Radar data is historically one of the most popular sources of information for nowcasting rainfall of storm cells [21], [22], [23], [24]. Traditional weather radar systems operate based on the emissions and receptions of horizontal pulses and corresponding Doppler effects [25]. They usually produce data products on variables associated with the reflectivity and velocity of objects in the atmosphere. In addition to the horizontal pulses, dual-polarimetric radar systems also operate with pulses in the vertical orientation. Unlike the case of the traditional weather radars, the two orientations of pulses allow the dual-polarimetric radars to produce data on more variables than the traditional radar systems for improved accuracy of estimates of rainfall [26], [27], [28].

The main objective of the study was to use ML regression methods to quantify the relative importance of radar variables for nowcasting heavy rainfall of a storm cell. For such a purpose, data products from a dual-polarimetric Doppler radar [29] for 17 variables for the Korean Peninsula were used for model calibration. Important input variables were then selected with a permutation importance algorithm to recalibrate ML regression models to compute the 30- and 60-min predictions of rainfall for MCS storm cells. The recalibrated ML regression models were applied to show their utility in providing early warnings for areas that would potentially experience heavy rainfall events. To do so, the recalibrated ML regression models were coupled with ML classification models that predict locations of storm cells. The ML classification models for location prediction were trained with data on locations of storm cells at now moments and the same 17 radar variables plus five topographic variables.

The rest of this article consists of four sections. The first section introduces the study area and data for assessing importance of radar variables and for ML modeling to nowcast, and to provide early warnings of, heavy rainfall. Then, the methods and algorithms to derive ML models and to apply these models for early warnings of heavy rainfall are delineated. Next, the results of implementing the ML methodology of the study are laid out and discussed. The final section concludes with summaries of the significance and limitations of the study and suggestions for future work.

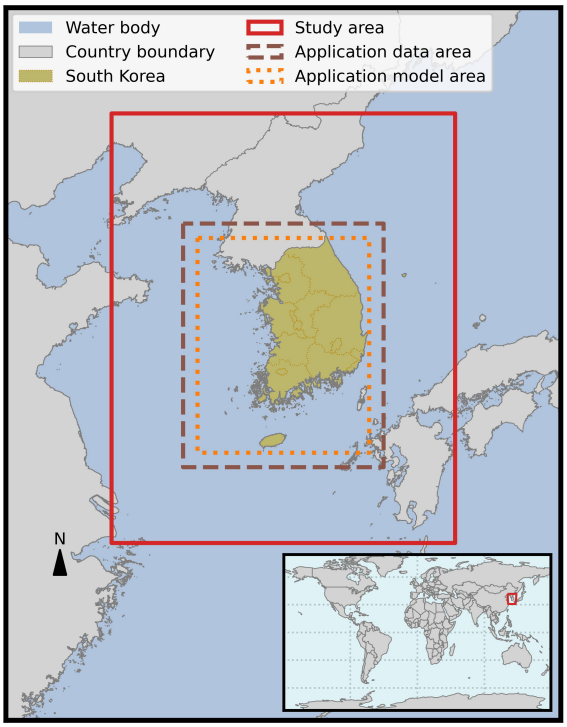


Fig. 1. Study area.

II. Data

A. Study Area

The study area of this research covers most of the Korean Peninsula (Fig. 1). It spans 960 km from west to east and 1200 km from north to south. The study area corresponds to the geographic extent of the data for assessing the relative importance of radar variables. As shown in Fig. 1, the data for application in rainfall nowcasting and provision of early warnings has a geographic extent indicated by the brown dashed lines. The orange dotted lines demarcate the model area for application in rainfall nowcasting and provision of early warnings. Shapefiles of country boundaries for visualization and data processing were downloaded from the database of global administrative areas (GADM) [30]. During the processing of spatial data, the Lambert conformal conic projection [31] was adopted along with a uniform spatial resolution of 1 km.

B. Data for Regression Modeling

Data on input variables for assessing their importance in nowcasting rainfall were based on 3-D composites from the dual polarimetric Doppler radar operated by the Korea Meteorological Administration (KMA). Each record of the data on input variables corresponded to a storm cell at a timestamp at the temporal resolution of 10 min. Storm cells were identified with the application of the fuzzy logic algorithm for storm tracking (FAST) [32]. Data on output variables included images of hybrid surface rainfall (HSR) in terms of rain rate at the spatial resolution of 1 km and temporal

resolution of 10 min. The HSR images were derived from the lowest observable data from individual radars and were then composited over their corresponding domains [33]. For each HSR image, two variables, i.e., mean rain rate and top 10% mean rain rate, were computed to be associated with the input variables at the storm cell level based on the areas of storm cells identified by the FAST algorithm. Mean rain rate referred to the average rain rate across the entire area of a storm cell. Top 10% mean rain rate was the average of rain rate among those pixelated areas with the highest 10% rain rate within the storm cell. Only data on MCS during the warm seasons from April to September were considered for model calibration. The time period of data was from April 2018 to September 2020.

Subsequent to using the FAST algorithm, identified storm cells were further categorized into four types, i.e., convective cell (CC), mesoscale convective complex (MCC), diagonal squall line (SLD), and parallel squall line (SLP), similar to classification by Lee and Kim 2007 [34]. Regarding each storm cell, if its longest radius, R_{mj}, was less than 20 km, it was classified as a CC. Among the non-CC storm cells, those with an axis ratio less than 4 were considered MCCs and the rest were identified as squall lines, where axis ratio referred to the ratio of the longest radius to the shortest radius. Among the squall lines, those with an angle between the longest radius and its advection larger than 45° were categorized as SLDs and the rest as SLPs.

For each storm cell across its lifespan, data points were created for training ML regression models for assessing relative importance of input variables. Each data point corresponded to the two output variables of mean rain rate (MeanRR) and top 10% mean rain rate (Top10%) at a now timestamp and 17 input variables (Table I) at a previous timestamp. For the nowcasting purpose, the previous timestamp was set to be either 30 or 60 min prior to the now timestamp. To facilitate storage and processing of data, each variable value was converted into an integer by multiplication with a factor to match the integer format of the time and identifier of the corresponding data point (Table I). To enhance the representativeness of highly positively skewed variables and to stabilize their variation, a base-10 logarithmic transformation was applied to these skewed variables (Table I).

Eight datasets were created for assessing relative importance of radar variables. Each dataset corresponded to 1 of 4 storm types and 1 of 2 prior times. For each dataset, each of the input and output variables was standardized with

$$v_S = \frac{v_O - \mu_O}{\sigma_O} \tag{1}$$

where v_S was the standardized variable, v_O was the variable after multiplication to integer and selective logarithmic transformation shown in Table I, and μ_O and σ_O were, respectively, the mean and standard deviation of v_O. With consideration of the timestamp of KST 2020-07-19 00:00:00 as the cutoff, the standardized datasets were then separated into the training datasets (before the cutoff timestamp) and validation datasets (at or after the cutoff timestamp). For deriving the permutation importance of input variables, each training dataset was further randomly separated into a training-training dataset (90% of training data points) and a training-testing dataset (the rest of the training data points). Because of limited numbers of data points for storm cell types of MCC, SLD, and SLP for each prior time, data points of these three storm types were pooled to form datasets for the joint MCC-SLD-SLP (MSL) storm type for importance assessment.

TABLE I

INPUT VARIABLES FOR ASSESSMENT OF RELATIVE IMPORTANCE IN RAINFALL PREDICTION

Variable	Description	Multiplication factor	Logarithmically transformed
R_{mj}	Longest radius in km	100	Yes
R_{mn}	Shortest radius in km	100	Yes
Theta	Direction of the shortest radius axis in degree	100	No
MeanZ	Storm area averaged reflectivity in dBZ	100	Yes
Area	Storm area in km^2	10	Yes
Volume	Storm area volume in km^3	10	Yes
Top	Echo top height in km	100	No
Base	Echo bottom height in km	100	No
MaxZ	Storm area maximum reflectivity in dBZ	100	Yes
$MaxZ_{hgt}$	Maximum reflectivity height in km	100	No
AvgVIL	Storm area averaged vertically integrated liquid water in km/m^2	100,000	Yes
MaxVIL	Storm area maximum vertically integrated liquid water in km/m^2	100,000	Yes
U	Zonal component of cell motion velocity in m/s	100,000	No
V	Meridional component of cell motion velocity in m/s	100,000	No
Direction	Storm propagation direction in degree	100,000	No
MeanRR	Mean rain rate in mm/h	100	Yes
Top10%	Mean rain rate of top 10% rain rate pixels in mm/h	100	Yes

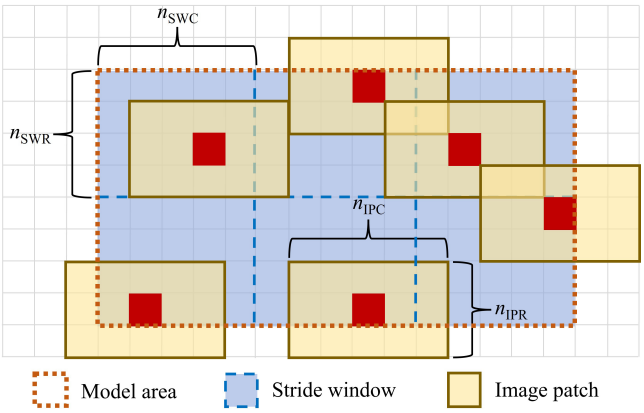


Fig. 2. Generation of image patches for training classification models for predicting storm cell locations. n_{IPR} and n_{IPC} indicate the size of image patch. n_{SWR} and n_{SWC} indicate the size of stride window. Each red square is a randomly selected central pixel of image patch.

C. Data for Classification Modeling

For predicting locations of storm cells, data in the format of image patches were generated (Fig. 2) to train ML

classification models to predict whether the central pixel of an image patch belongs to a storm cell after a considered time period of either 30 or 60 min. Such an image patch showed the geographic distribution of a storm cell around its central pixel. The output data for classification modeling was the binary variable of whether the central pixel of an image patch belonged to a storm cell of interest at a now timestamp. The input data included the image patches of the same binary variable at all pixels of each image patch plus 22 variables at the central pixel of the image patch at a previous timestamp.

As shown in Fig. 2, each image patch had the size of n_{IPR} by n_{IPC}, where n_{IPR} and n_{IPC} were, respectively, the numbers of rows and columns of the image patch. Stride windows of the size of n_{SWR} by n_{SWC} were created, where n_{SWR} and n_{SWC} were, respectively, the numbers of rows and columns of a stride window. Within each stride window, one pixel was randomly selected (highlighted in red color) as the targeted pixel. Around this targeted pixel, an image patch was then retrieved given n_{IPR} and n_{IPC}. Each pixel of an image patch was assigned a value of 1 or 0 if it belonged to a storm cell of interest or not. Image patches with all pixels of 0s were discarded from model training and validation. In this study, $n_{\mathrm{IPR}}= 81$, $n_{\mathrm{IPC}}= 81$, $n_{\mathrm{SWR}}= 10$, and $n_{\mathrm{SWC}}= 10$. These parametric values were determined based on trials and errors to provide a good model performance within an acceptable computational time.

Among the 22 input variables for classification modeling, 17 were the same as the input variables for regression modeling (Table I). The other five were topographic variables derived with the open-source software QGIS 3.20.3-Odense [35] based on the spatial data used in the study. For computing the topographic variables, the digital elevation model (DEM) dataset of Global Multi-Resolution Terrain Elevation Data 2010 at the spatial resolution of 7.5 arc-seconds was downloaded from the United States Geological Survey website [36]. The created topographic variables included distance to sea, elevation, aspect, roughness, and slope.

The same cutoff timestamp, i.e., KST 2020-07-19 00:00:00, as of the data for assessing variable importance was used to separate the training and validation datasets for ML classification modeling. A total of 12 datasets were created for training classification models. Half were for the 30-min modeling and the other half were for the 60-min modeling. For either prior time, the six datasets included ones for CC, MCC, SLD, SLP, MSL, and all storm cell types pooled together (All), respectively.

III. Methodology

The presented study consists of two parts. The first part assessed the relative importance of radar variables in nowcasting heavy rainfall by adopting supervised ML regression techniques. The developed ML regression models could be used to predict the mean rain rate and top 10% mean rain rate of an MCS storm cell. The second part of the study developed supervised ML classification models based on a CNN modeling method to predict storm cell locations and applied the results of regression modeling of the first part to produce nowcasts of heavy rainfall conditions and provide early warnings. Quantitative models in this study were established with the open-source software language Python 3.8.11 [37].

A. Regression Modeling

To assess the relative importance of radar variables in nowcasting heavy rainfall, four ML regression methods were adopted. The initial input of the ML regression models included 17 variables listed in Table I. The output variables were the mean rain rate and top 10% mean rain rate. The adopted ML regression modeling methods included the least absolute shrinkage and selection operator (Lasso) [38], [39], random forest regression (RFR) [40], [41], support vector regression (SVR) [42], and artificial neural network (ANN) [43]. After each ML model was calibrated with its training-training dataset, a permutation importance metric was computed for each input variable on the corresponding training-testing dataset. Based on the derived metrics of permutation importance, pertinent input variables were selected. The ML regression model with the selected input variables was then recalibrated with its entire training dataset. The recalibrated ML regression model was finally tested on its validation dataset.

1) Lasso: The Lasso method is a linear regression approach that also selects variables and performs regularization simultaneously. Consider a linear regression model

$$\boldsymbol{y}_{\mathrm{L}} = \boldsymbol{X}_{\mathrm{L}}\boldsymbol{\beta}_{\mathrm{L}} + \sigma_{\mathrm{L}}\boldsymbol{\varepsilon}_{\mathrm{L}} \tag{2}$$

where $\boldsymbol{y}_{\mathrm{L}}$ is the column vector of values of an output variable, $\boldsymbol{X}_{\mathrm{L}}$ is the data matrix of input variables, $\boldsymbol{\beta}_{\mathrm{L}}$ is a column vector of model coefficients, $\boldsymbol{\varepsilon}_{\mathrm{L}}$ is a column vector of independent and identically distributed standard normal random variables, and σ_{L} is the dispersion parameter of the model. Lasso estimates

$$\hat{\beta}_{\mathrm{L}} = \left(\left\| \boldsymbol{y}_{\mathrm{L}} - \boldsymbol{X}_{\mathrm{L}}\boldsymbol{\beta}_{\mathrm{L}} \right\|^2 + \lambda_{\mathrm{L}} \sum_{h=1}^{m} |\beta_{\mathrm{L}h}| \right) \tag{3}$$

where $\hat{\beta}_{\mathrm{L}}$ is the Lasso estimator of $\boldsymbol{\beta}_{\mathrm{L}} = [\beta_{\mathrm{L}0}, \beta_{\mathrm{L}1}, \ldots, \beta_{\mathrm{L}m}]^{\mathrm{T}}$, $\beta_{\mathrm{L}0}$ is the intercept, $\beta_{\mathrm{L}h}$ is the hth element of $\boldsymbol{\beta}_{\mathrm{L}}$, m is the number of input variables, λ_{L} is a nonnegative regularization parameter, $\|\cdot\|$ is the L2 norm operator, and $|\cdot|$ is the absolute value operator [38], [39]. The *LassoCV* function from the scikit-learn library was adopted for Lasso modeling [44].

2) Random Forest Regression: RFR is a regression modeling approach that randomly establishes regression trees and uses the mean output of the regression trees as the estimate of target value of output given the values of input variables [40], [41]. The RFR algorithm used in this study consisted of four steps. First, given a training-training dataset with the sample size of n, n samples were randomly selected from the training-training dataset with replacement to form a new dataset. Second, a regression tree was established with the new dataset. Third, the first and second steps were repeated for 800 times. Fourth, given the input values of a new data point, the outputs of the established 800 regression trees were averaged to produce the estimate of the output of the RFR

model. The RFR was achieved via the *RandomForestRegressor* function of the scikit-learn library [44].

3) Support Vector Regression: SVR is a computational approach that produces a regression model in the form of a hyperplane in the original or transformed vector space of data points involving both the input and output variables. The hyperplane is at the center between two parallel margins that contain most of the data points and touched only by data points called support vectors. For a linear SVR model, its training-training dataset with sample size n can be denoted as

$$\{(\boldsymbol{x}_{S1}, y_{S1}), (\boldsymbol{x}_{S2}, y_{S2}), \ldots, (\boldsymbol{x}_{Sn}, y_{Sn})\} \subset X \times R \tag{4}$$

where X is the vector space of input variables. The SVR hyperplane is

$$y_S = f_S(\boldsymbol{x}_S) = \langle \boldsymbol{w}_S \mid \boldsymbol{x}_S \rangle + b_S \tag{5}$$

where $\boldsymbol{w}_S \in X$, $b_S \in R$, and $\langle \cdot \mid \cdot \rangle$ refers to the dot product in X. $f_S(\boldsymbol{x}_S)$ can be derived by minimizing

$$L_S = \frac{1}{2}\|\boldsymbol{w}_S\|^2 + C_S \sum_{i=1}^{n} (\xi_{Si} + \zeta_{Si}) \tag{6}$$

subject to

$$\begin{cases} y_{Si} - \langle \boldsymbol{w}_S \mid \boldsymbol{x}_{Si} \rangle - b_S \le \varepsilon_S + \xi_{Si} \\ \langle \boldsymbol{w}_S \mid \boldsymbol{x}_{Si} \rangle + b_S - y_{Si} \le \varepsilon_S + \zeta_{Si} \\ \xi_{Si}, \zeta_{Si} \ge 0 \end{cases} \tag{7}$$

where ε_S is the precision parameter of the support vector margins, $C_S > 0$ is a parameter determining the flatness of model hyperplane and the toleration of deviations of data points from the margins, and ξ_{Si} and ζ_{Si} are, respectively, the ith values of two slack variables [42]. A slack variable indicates the distance, from either side of the model hyperplane, between a data point and its closer support vector margin. The introduction of slack variables allows model calibration with data points beyond the support vector margins. In this study, parameters were set at the most commonly used values, as $C_S = 1$ and $\varepsilon_S = 0.01$.

The optimization problem of (6) and (7) can be solved via the method of Lagrange multipliers with a focus on the computation of $\langle \boldsymbol{x}_{Si} \mid \boldsymbol{x}_{Sj} \rangle$ for all data points, where i and j are dummy variables referring to the ith and jth data points, respectively. With a kernel trick

$$K_S(\boldsymbol{x}_{Si}, \boldsymbol{x}_{Sj}) = \langle \phi(\boldsymbol{x}_{Si}) \mid \phi(\boldsymbol{x}_{Sj}) \rangle \tag{8}$$

where $\phi(\cdot)$ is a mathematical mapping, the original input variables of $\boldsymbol{x}_S$ can be transformed into a new set of variables in a new vector space with higher dimensions than X. Such a transformation of input variables can help solve a nonlinear regression problem. In addition to the linear SVR approach, in this study, a radial basis function (RBF) kernel was adopted as

$$K_{SR}(\boldsymbol{x}_{Si}, \boldsymbol{x}_{Sj}) = \exp\left(-\gamma_S \|\boldsymbol{x}_{Si} - \boldsymbol{x}_{Sj}\|^2\right) \tag{9}$$

where

$$\gamma_S = \frac{n}{\sum_{i=1}^{n} \sum_{j=1}^{m} (x_{Sij} - \bar{x}_S)^2} \tag{10}$$

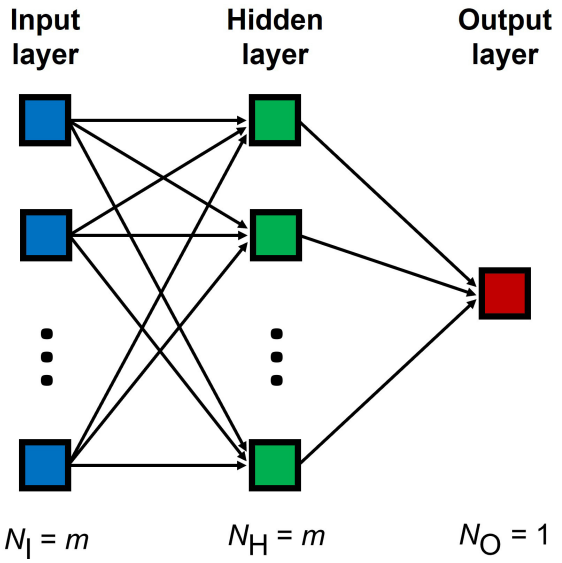


Fig. 3. MLP for assessing importance of input variables in nowcasting heavy rainfall.

was a common configuration. With a Taylor expansion, $K_{SR}(\cdot)$ can be shown to be the dot product of mappings, $\phi(\cdot)$, of input variables from the original vector space into a vector space with an infinite number of dimensions. The SVR function from the scikit-learn library was used to apply the linear and RBF SVR approaches [44].

4) Artificial Neural Network: ANN is a computational model that mimics the biological neural network of an animal brain to produce output based on input values. The generic architecture of ANN, the MLP [43], was adopted for assessing importance of radar variables in this study. For each training-training dataset, two MLPs were established for the two output variables, respectively. Each MLP consisted of one input layer with $N_I = m$ nodes corresponding to the m input variables, one hidden layer with $N_H = m$ nodes, and one output layer with $N_O = 1$ node corresponding to the output variable (Fig. 3). Each node of the hidden and output layers performed

$$x_{Aq,r} = f_{q,r}\left(\beta_{q,r,0} + \sum_{s=1}^{N_{q-1}} \beta_{q,r,s} x_{Aq-1,s}\right) \tag{11}$$

where $x_{Aq,r}$ referred to the output of the rth node of the qth layer, $f_{q,r}(\cdot)$ was an activation function, $\beta_{q,r,0}$ was a bias parameter, $\beta_{q,r,s}$ was a weight parameter, and N_{q-1} was the number of nodes of the layer previous to the qth layer. The rectified linear unit (ReLU) function [45], [46]

$$f_{ReLU}(x) = \max(0, x) \tag{12}$$

and the identical functions were used, respectively, as the activation functions for the hidden and output layers. Each MLP was trained with 400 epochs with the adaptive moment estimation (Adam) algorithm [47] with the batch size of 16 and the initial learning rate at 2×10^{-4}. The mean squared error (MSE) was adopted as the loss function

$$\text{MSE} = \frac{1}{n_y} \sum_{i=1}^{n_y} (y_i - \hat{y}_i)^2 \tag{13}$$

where n_y was the number of data points, y_i was the observed output value of the ith data point, and $\hat{y}_i$ was its model prediction. The ANN modeling was achieved via the TensorFlow library [48].

5) Variable Selection: To determine the relative importance of input variables of each ML regression model, an algorithm based on the permutation importance technique was applied [49]. The adopted algorithm also selected pertinent variables for each model given their computed permutation importance scores. The permutation importance score was associated with the change in a model performance when the values of a single input variable were randomly shuffled. The algorithm used in this study goes as follows. Step 1 calibrates an ML regression model on its corresponding training-training dataset. Step 2 computes the root MSE (RMSE) of the model on its training-testing dataset as a reference score RS. Step 3 selects the first input variable or the next input variable if the first has already been selected. Step 4 randomly shuffles the values of the selected input variable to generate a corrupted version of the training-testing dataset. Step 5 computes the RMSE of the model on the corrupted training-testing dataset as the variable shuffle score VSS. Step 6 derives the importance score IS $=$ VSS $-$ RS and records it. Step 7 repeats steps 4–6 100 times to record 100 ISs. Step 8 continues steps 3–7 to cover all the input variables. Step 9 conducts one-tailed one-sample t-tests on ISs for the input variables with the null hypothesis that the actual IS of each of the input variables is no greater than 0. Step 10 selects the input variables for which the null hypothesis is rejected at the significance level of 10^{-9}. Step 11 recalibrates the ML regression model with the selected input variables on the entire training dataset including the training-training and training-testing datasets.

6) Validation: For each recalibrated ML regression model, a validation was conducted on its validation dataset. For validation, model performances on predictions of mean rain rate and top 10% mean rain rate were compared to the results by directly using the prior values (i.e., the input variables) to predict the current values of the two output variables. Four loss functions were used to indicate model performance. They included the mean absolute error (MAE) (14), the mean absolute percentage error (MAPE) (15), MSE (13), and the coefficient of determination (R^2) (16)

$$\text{MAE} = \frac{1}{n_y}\sum_{i=1}^{n_y}|y_i - \hat{y}_i| \tag{14}$$

$$\text{MAPE} = \frac{1}{n_y}\sum_{i=1}^{n_y}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \tag{15}$$

$$R^2 = 1\frac{\sum_{i=1}^{n_y}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n_y}(\bar{y} - \hat{y}_i)^2} \tag{16}$$

where $\bar{y}$ was the mean of observed values of output variable.

B. Application

Subsequent to the assessment of relative importance of input variables including radar products in nowcasting heavy rainfall with ML regression models, the recalibrated ML regression models were further applied to map areas with potential

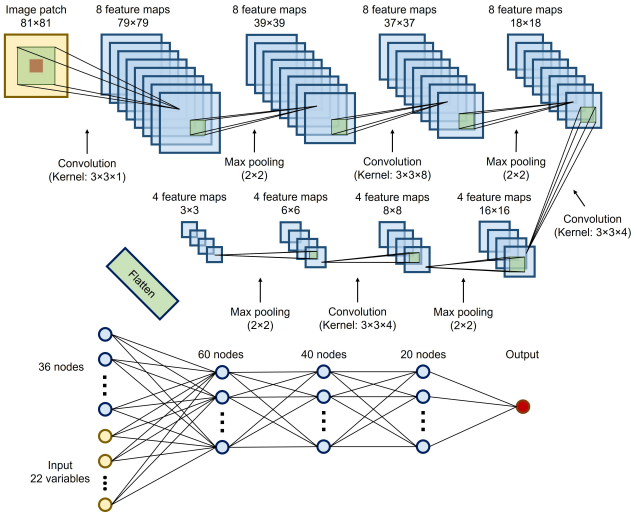


Fig. 4. Architecture of the CNN-based machine learning classification model.

of heavy rainfall in 30 and 60 min given values of input variables at current time. Such an application was achieved by integrating an ML regression modeling result with an ML classification model that predicted the location of a storm cell of interest.

1) Classification Modeling: The ML classification models used in this study were to classify whether a pixel on an image, corresponding to a location, would belong to a storm cell of interest in 30 or 60 min. The input of the ML classification models included 17 variables listed in Table I plus five topographic variables, i.e., distance to sea, elevation, aspect, roughness, and slope, at a prior timestamp. The output was the binary variable indicating if a storm cell of interest was covering the central pixel of the image patch at a now timestamp. To establish such a model, a 2-D CNN-based method was used. CNN is an ANN that uses mathematical convolution, instead of matrix multiplication, in at least one of its layers [50]. The convolutional responses are known as feature maps. For this application, a total of 12 classification models were trained in accordance with the CC, MCC, SLD, SLP, MSL, and All datasets given two prior times of 30 and 60 min.

The architecture of the ML classification models consisted of two input layers, four pairs of 2-D convolutional and max pooling layers, three fully connected dense layers, and one output layer (Fig. 4). The input layers included one layer of image patch and another layer of 22 input nodes corresponding to 17 variables listed in Table I and five topographic variables. After computations at four pairs of convolutional and max pooling layers on an input image patch, four feature maps of the size of 3 by 3 were flattened to form 36 nodes that were further concatenated with the 22 input nodes. The output layer produced an expected probability that the central pixel of the input image patch belonged to the storm cell of interest. To derive such an expected probability, the logistic sigmoid activation function was used for the output layer

$$f_{\text{LS}}(x) = \frac{\exp(x)}{\exp(x)+1}. \tag{17}$$

Apart from the output layer, each node of a computational layer used the exponential linear unit (ELU) function as its activation function [51]

$$f_{\mathrm{ELU}}(x) = \begin{cases} x & x > 0 \\ \exp(x) - 1 & x \leq 0. \end{cases} \tag{18}$$

The CNN-based classification model contained a total of 7925 trainable parameters for model calibration.

To calibrate the ML classification models, a shallow training scheme was adopted. According to this scheme, an entire model training process for each ML classification model involved 80 iterations. During each iteration, 20% of the one data points, i.e., data points with their output values equal to 1 indicating that the central pixel belonged to the storm cell of interest, were randomly selected from the training dataset without replacement. The same amount of 0 data points, i.e., data points with their output values equal to 0 indicating that the central pixel did not belong to the storm cell of interest, were randomly selected from the training dataset without replacement to match the number of selected 1 data points. The selected 1 and 0 data points were pooled to form a dataset. The pooled dataset was then used to train the same ML classification model with five epochs with the batch size of 64 before starting the next iteration. For training ML classification models, the Adam algorithm [47] was used for model calibration with the initial learning rate set at 5×10^{-4}. To facilitate binary classification, the binary cross entropy (BCE) was used as the loss function

$$\mathrm{BCE} = \frac{1}{n_y} \sum_{i=1}^{n_y} [y_i \ln(\hat{y}_i) + (1 - y_i) \ln(1 - \hat{y}_i)]. \tag{19}$$

A probability threshold of 0.5 was adopted to convert the final output into a binary format.

For validation, the trained ML classification models were applied to their corresponding validation datasets. Four metrics, including the false positive rate (FPR) (20), precision (Prec) (21), recall (Reca) (22), and F1 score (F1) (23), were computed to indicate model performance

$$\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TN}} \tag{20}$$

$$\mathrm{Prec} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{21}$$

$$\mathrm{Reca} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{22}$$

$$\mathrm{F1} = \frac{2\mathrm{PrecReca}}{\mathrm{Prec} + \mathrm{Reca}} \tag{23}$$

where TP, FP, TN, and FN referred to the numbers of true positives, false positives, true negatives, and false negatives, respectively.

To predict the geographical distribution of a storm cell of interest, an image patch, of the same size of a training image patch, around each pixel of the image were created for a now moment. If an image patch only contained 0s, the state of its central pixel would be predicted to be 0 in the future. Otherwise, the image patch along with its associated values of 22 input variables was fed into the corresponding ML classification model to derive an expected probability of its central pixel being part of the storm cell of interest in the future. The probability threshold of 0.5 was then applied such that the central pixel was considered as part of the storm cell if its output value was greater than or equal to 0.5 and not part of the storm cell if otherwise.

2) Nowcasting Heavy Rainfall: The end products of the application of the trained ML models were maps showing the areas with potential heavy rainfall in 30 or 60 min. To achieve this, the ML regression models for assessing variable importance were coupled with the ML classification models. Given values of input variables at a now moment, the ML classification model predicted the area of storm cell at a future timestamp. In the meantime, the ML regression models estimated the rain rate at the same future timestamp. A warning was then issued for the storm cell area, if the predicted top 10% mean rain rate was greater than or equal to 30 mm/h.

IV. Results and Discussion

Subsequent to delineation of the methodology of this study, this section displays and discusses the results of implementation of the proposed methodology. These results include the quantification of importance of input variables of ML regression models for nowcasting heavy rainfall, the validation of ML regression modeling for variable importance, the validation of ML classification modeling for predicting locations of storm cells, and an example of application of ML regression and classification models to issue early warning for an area with potential of experiencing heavy rainfall.

A. Variable Importance

With the permutation importance algorithm, 100 results of the relative importance of each input variable of the ML regression models in nowcasting heavy rainfall can be derived and displayed through box plots. Fig. 5 shows an example of such a box plot for the ANN model for top 10% mean rain rate of CC dataset with the 60-min forecasting time. The horizontal axis indicates the importance score IS. For comparison of variable importance, the mean values of variable importance were computed and ranked for each model (see Table II for nowcasting mean rain rate with CC dataset and 30-min forecasting time). For each combination of output variable, storm cell type, and forecasting time, the sum of the mean importance of each input variable was derived across all five ML regression models. The sums of ranks of input variables across the ML regression models were then ranked to indicate the importance of input variables across the adopted ML regression models. These total rankings of the input variables were finally listed in Table III across different models.

In Table III, the column of CC mean rain rate 30 min corresponded to the column of total rank in Table II. The other columns of Table III were the results of total ranks from other tables of rankings of permutation importance produced in the study. Here, as an example, the storm area averaged reflectivity (MeanZ) was shown to be more important in general than the direction of the shortest radius axis (Theta), as the total ranks

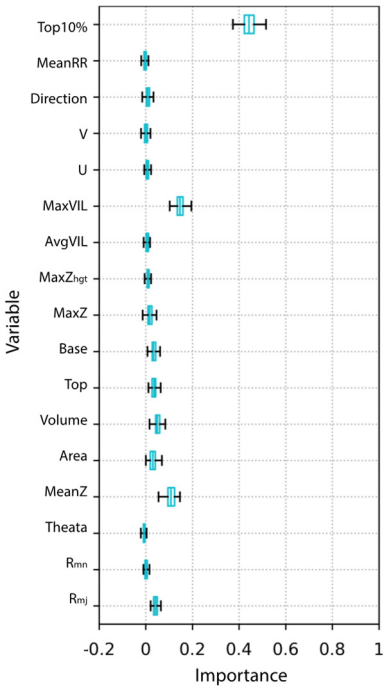


Fig. 5. Box plot of permutation importance of input variables of the ANN model for nowcasting top 10% mean rain rate with the CC dataset and the forecasting time of 60 min.

TABLE II

EXAMPLE OF RANKINGS OF PERMUTATION IMPORTANCE OF INPUT VARIABLES BY MODELS

Variable	Lasso	RFR	Linear SVR	RBF SVR	ANN	Sum	Total Rank
MeanRR	1	1	1	1	1	5	1
Volume	2	6	2	6	6	22	2
MaxVIL	3	9	3	4	4	23	3
MeanZ	5	11	4	3	3	26	4
Base	7	5	6	5	7	30	5
R_{mj}	6	3	5	11	8	33	6
Top10%	12	2	16	2	5	37	7
Area	4	13	15	12	2	46	8
R_{mn}	14	7	7	10	9	47	9
MaxZ	11	8	10	9	10	48	10
AvgVIL	8	4	9	14	13	48	10
$MaxZ_{hgt}$	15	10	8	8	15	56	12
Top	13	15	14	7	11	60	13
U	10	16	12	17	12	67	14
V	9	12	17	13	16	67	14
Theta	16	14	13	15	17	75	16
Direction	17	17	11	16	14	75	16

TABLE III

TOTAL RANKINGS OF PERMUTATION IMPORTANCE OF INPUT VARIABLES FOR NOWCASTING RAINFALL

Variable	CC				MSL			
	Mean Rain Rate		Top 10% Mean Rain Rate		Mean Rain Rate		Top 10% Mean Rain Rate	
	30 Min	60 Min	30 Min	60 Min	30 Min	60 Min	30 Min	60 Min
R_{mj}	6	7	7	6	8	4	7	4
R_{mn}	9	7	12	13	4	5	10	8
Theta	16	16	14	16	16	16	16	11
MeanZ	4	3	4	3	2	6	3	6
Area	8	12	8	8	10	8	8	9
Volume	2	3	5	5	6	3	6	3
Top	13	10	10	7	13	10	5	10
Base	5	6	5	4	7	11	11	13
MaxZ	10	15	9	9	8	7	8	5
$MaxZ_{hgt}$	12	13	13	11	15	12	14	15
AvgVIL	10	9	11	12	11	15	12	14
MaxVIL	3	3	3	2	5	9	4	7
U	14	14	15	15	17	14	17	15
V	14	11	15	10	12	17	13	17
Direction	16	17	17	17	14	13	15	11
MeanRR	1	1	2	13	1	1	2	2
Top10%	7	2	1	1	3	2	1	1

TABLE IV

TOTAL RANKINGS OF SELECTIONS OF INPUT VARIABLES FOR NOWCASTING RAINFALL

Variable	CC				MSL			
	Mean Rain Rate		Top 10% Mean Rain Rate		Mean Rain Rate		Top 10% Mean Rain Rate	
	30 Min	60 Min	30 Min	60 Min	30 Min	60 Min	30 Min	60 Min
R_{mj}	1	6	1	7	4	5	1	1
R_{mn}	11	6	11	11	4	1	10	11
Theta	17	17	13	16	16	13	14	6
MeanZ	1	1	1	1	1	5	5	1
Area	8	13	8	11	9	9	5	6
Volume	1	1	1	1	9	1	10	1
Top	11	6	8	7	13	9	5	14
Base	1	6	1	1	1	13	5	11
MaxZ	8	13	11	1	4	5	5	6
$MaxZ_{hgt}$	8	6	13	7	13	13	14	17
AvgVIL	1	6	8	11	9	9	10	6
MaxVIL	1	1	1	1	9	5	1	6
U	15	13	16	11	16	9	17	16
V	11	6	13	7	4	16	10	14
Direction	15	16	17	16	13	16	14	11
MeanRR	1	1	1	15	1	1	1	1
Top10%	11	1	1	1	4	1	1	1

of storm area averaged reflectivity were consistently much smaller than the ones of direction of the shortest radius axis in Table III.

In addition to the mean importance score, whether input variables were selected as pertinent predictors can also be used to indicate the relative importance of the input variables (see Table S1 for nowcasting mean rain rate with CC dataset and 30-min forecasting time). In Table S1, a selection score, either 1 or 0, was assigned to a variable (a row) and a model (columns Lasso to ANN) if the variable was selected by the model or not, respectively. Then, the sums of selection scores were computed and ranked. Table IV lists the total rankings of the sum of selection scores of input variables across different models.

In Table IV, for example, the column of CC mean rain rate 30 min recorded the result of the column of total rank in Table S1. The other columns of Table IV corresponded to the columns of total rank of other tables of rankings of variable selection generated in the study. Here, similar to Table III, Table IV also indicated that the storm area averaged reflectivity

was more important in general than the direction of the shortest radius axis, as the total ranks of variable selection of storm area averaged reflectivity were consistently much smaller than the ones of direction of the shortest radius axis.

For nowcasting mean rain rate for CC cells with 30-min input, radar variables of storm area volume, storm area maximum vertically integrated liquid water, storm area averaged reflectivity, echo bottom height, and the longest radius were identified as pertinent predictors (Tables III and IV). With input of 60 min prior, radar variables of storm area averaged reflectivity, storm area volume, and storm area maximum vertically integrated liquid water were determined to be important predictors of mean rain rate for CC cells (Tables III and IV).

For nowcasting the output variable of top 10% mean rain rate, the list of the most important radar variables was similar to the one for mean rain rate for CC storm cells. The 30-min models identified the storm area maximum vertically integrated liquid water, storm area averaged reflectivity, storm area volume, echo bottom height, and the longest radius as the pertinent predictors (Tables III and IV). The 60-min models highlighted the storm area maximum vertically integrated liquid water, storm area averaged reflectivity, echo bottom height, and storm area volume as the most important radar variable predictors (Tables III and IV).

With the MSL datasets, the calibration of ML regression models resulted in a different list of the most pertinent radar variable predictors of mean rain rate as with the CC datasets. In particular, the shortest radius was identified as a highly important predictor by the ML regression models calibrated with the MSL datasets (Tables III and IV). In addition to the shortest radius, the 30-min models with MSL datasets identified the storm area averaged reflectivity, echo bottom height, the longest radius, and the storm area maximum reflectivity as the most pertinent radar variable predictors, despite that the rankings and selections of the input variables were not perfectly consistent with each other. On the other hand, the 60-min models produced consistent results for both rankings and selections of input variables, as they identified the storm area volume, longest radius, shortest radius, storm area averaged reflectivity, and storm area maximum reflectivity as the most important radar variable predictors.

For nowcasting the top 10% mean rain rate with the MSL datasets, the 30-min models identified the storm area averaged reflectivity, storm area maximum vertically integrated liquid water, and echo top height as the important radar variables (Tables III and IV). Meanwhile, the 60-min models highlighted storm area volume, longest radius, storm area maximum reflectivity, and storm area averaged reflectivity as the pertinent radar variable predictors of top 10% mean rain rate (Tables III and IV).

According to the ranking aggregations, the most commonly identified pertinent radar input variables were storm area averaged reflectivity and storm area volume. Then, radar variables of storm area maximum vertically integrated liquid water, echo bottom height, and longest radius were also highlighted as highly important predictors. The identification of these radar variables as important predictors of mean rain rate and top 10% mean rain rate echoed the rationale for using products on these radar variables to model and indicate rainfall distributions. On the other hand, the radar variables examined to be of medium importance included shortest radius and storm area maximum reflectivity. The less important radar variables were echo top height, storm area, and storm area averaged vertically integrated liquid water. Last, the radar variables ranked as the least important for nowcasting heavy rainfall included the maximum reflectivity height, zonal component of cell motion velocity, meridional component of cell motion velocity, direction of the shortest radius axis, and storm propagation direction.

Here, interestingly, the directions and velocities of a storm cell were considered the least important predictors. Regarding the directions of storm cells, because the convective storms recorded in the data tended to have similar directions, these variables did not provide information with sufficient variation for the ML regression models to predict rainfall. Meanwhile, although velocities of storm cells were considered trivial in nowcasting rain rate in our study, they should still be treated as important factors in flood management, as slow-moving heavy storms tend to result in flood conditions.

The rankings of highly important input variables based on aggregation across all ML regression models look similar to a large extent in general. Given the same storm cell type and prior time, the rankings of important radar variables for nowcasting mean rain rate and top 10% mean rain rate were similar. Likewise, for the same output variable with the same dataset, ML regression models of different prior times of input also produced similar rankings of input variables. Regarding different storm cell types, however, the rankings for CC and MSL datasets did differ from each other, given the same output variable and prior time.

Despite the similar identifications of relative importance of radar variables and the other input variables in general, each individual ML regression model tended to rank the variable importance in a slightly different manner. For example, for nowcasting top 10% mean rain rate with CC dataset and 60-min input values, although the Lasso, linear SVR, and ANN models all identified the storm area volume as highly important, both the RFR and RBF SVR models ranked this input variable as the 12th important. Such a difference may be associated with the capabilities of the corresponding ML regression models to handle nonlinearities of input variables. In this study, the Lasso and linear SVR models were linear, while RFR and RBF SVR models were nonlinear. Meanwhile, the MLP ANN model with only one hidden layer was close to a liner model. Nevertheless, the differences between the rankings of variable importance for individual ML regression models were small.

In this study, although we used a dual-polarimetric radar to collect data on input variables of ML regression models, information on these variables can also be obtained via a conventional weather radar. We did not include data produced exclusively by a dual-polarimetric radar because our data should be consistent with the variables used for the FAST algorithm, while the FAST algorithm only required data that could be produced by a conventional weather radar. Future work needs to examine to what extent additional

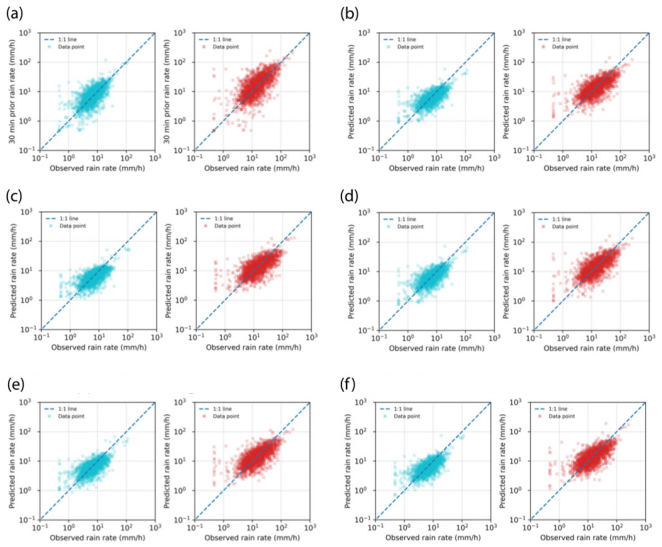


Fig. 6. Performance of models with 30-min data for CC storm cells. (a) Benchmark MeanRR (left) and Top10% (right). (b) Lasso MeanRR (left) and Top10% (right). (c) RFR MeanRR (left) and Top10% (right). (d) Linear SVR MeanRR (left) and Top10% (right). (e) RBF SVR MeanRR (left) and Top10% (right). (f) ANN MeanRR (left) and Top10% (right).

variables generated only by a dual-polarimetric radar, such as differential reflectivity and specific differential phase, may improve the predictive performance of ML regression models for nowcasting rainfall.

B. Regression Modeling Validation

For validation of ML regression models for nowcasting mean rain rate and top 10% mean rain rate, the model performance was compared to a benchmark, where the rain rate of storm cell was matched with the time delay of forecasting time of 30 or 60 min (e.g., 30-min prior mean rain rate versus current mean rain rate). The benchmark represented the accuracy of perfect advection forecasting without ML regression models. As an example, Fig. 6(a) shows the results of the benchmark for CC storm cells with input values of 30 min prior. For a perfect model, the data points would align along the one-to-one dashed lines. For CC storm cells, Fig. 6(b)–(f) displays the validation results of the 30-min forecasting with the ML regression models. For CC storm cells, the ML regression models adopted in the study resulted in better validation performance than the benchmark model for most of the cases (see Tables V and VI). In particular, the RBF SVR and ANN models provided the best prediction results with the 60- and 30-min input values, respectively.

For MSL storm cells, as another example, Fig. 7(a) displays the performance results of benchmark, comparing the current values of output variables with the values of 30 min prior. The validation results of ML regression models for 30-min nowcasting are presented in Fig. 7(b)–(f). For most of the cases for MSL storm cells, the adopted ML regression models produced better validation performance than the benchmark model (see Tables VII and VIII). Apart from RFR, all the other ML regression models resulted in the best performance on some occasions. Regarding the coefficient of determination,

TABLE V

VALIDATION PERFORMANCE FOR MEAN RAIN RATE BY MODELS WITH 30-MIN CC DATASET

Model	MAE	MAPE	MSE	R^2
Benchmark	3.14×10^{2}	5.35×10^{-1}	2.62×10^{5}	4.91×10^{-1}
Lasso	*2.88×10^{2}*	***4.35×10^{-1}***	2.62×10^{5}	4.91×10^{-1}
RFR	*2.90×10^{2}*	*4.39×10^{-1}*	*2.38×10^{5}*	*5.36×10^{-1}*
Linear SVR	*2.86×10^{2}*	*4.57×10^{-1}*	*2.37×10^{5}*	*5.39×10^{-1}*
RBF SVR	*2.89×10^{2}*	*4.49×10^{-1}*	*2.53×10^{5}*	*5.07×10^{-1}*
ANN	***2.86×10^{2}***	*4.39×10^{-1}*	***2.17×10^{5}***	***5.77×10^{-1}***

Note: *Italic* indicates better performance than benchmark model; **bold** indicates best performance regarding a metric.

TABLE VI

VALIDATION PERFORMANCE FOR TOP 10% MEAN RAIN RATE BY MODELS WITH 30-MIN CC DATASET

Model	MAE	MAPE	MSE	R^2
Benchmark	9.62×10^{2}	8.42×10^{-1}	2.37×10^{6}	3.50×10^{-1}
Lasso	*8.45×10^{2}*	***6.40×10^{-1}***	*1.97×10^{6}*	*4.60×10^{-1}*
RFR	*8.48×10^{2}*	*6.42×10^{-1}*	*1.90×10^{6}*	*4.78×10^{-1}*
Linear SVR	*8.46×10^{2}*	*6.93×10^{-1}*	*1.91×10^{6}*	*4.76×10^{-1}*
RBF SVR	***8.40×10^{2}***	*6.72×10^{-1}*	*1.86×10^{6}*	*4.90×10^{-1}*
ANN	*8.43×10^{2}*	*6.58×10^{-1}*	***1.81×10^{6}***	***5.03×10^{-1}***

Note: *Italic* indicates better performance than benchmark model; **bold** indicates best performance regarding a metric.

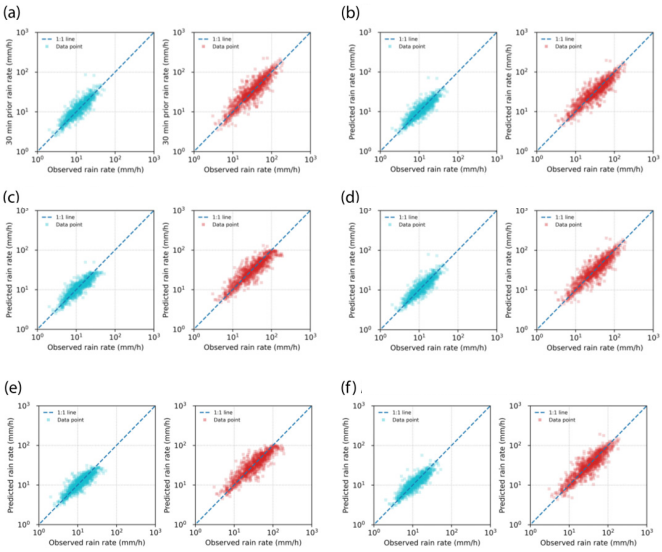


Fig. 7. Performance of models with 30-min data for MSL storm cells. (a) Benchmark MeanRR (left) and Top10% (right). (b) Lasso MeanRR (left) and Top10% (right). (c) RFR MeanRR (left) and Top10% (right). (d) Linear SVR MeanRR (left) and Top10% (right). (e) RBF SVR MeanRR (left) and Top10% (right). (f) ANN MeanRR (left) and Top10% (right).

for example, the RBF SVR, ANN, linear RBF, and Lasso models produced, respectively, the best results for 30-min mean rain rate, 60-min mean rain rate, 30-min top 10% mean rain rate, and 60-min top 10% mean rain rate. These validation results suggested that inclusion of radar variables in ML regression models was effective and significant in improvement of model performance for nowcasting mean rain

TABLE VII

VALIDATION PERFORMANCE FOR MEAN RAIN RATE BY MODELS WITH 30-MIN MSL DATASET

Model	MAE	MAPE	MSE	R^2
Benchmark	2.50×10^{2}	1.98×10^{-1}	2.06×10^{5}	5.78×10^{-1}
Lasso	*2.40×10^{2}*	*1.86×10^{-1}*	*1.64×10^{5}*	*6.63×10^{-1}*
RFR	*2.46×10^{2}*	*1.86×10^{-1}*	*1.49×10^{5}*	*6.96×10^{-1}*
Linear SVR	***2.38×10^{2}***	***1.85×10^{-1}***	*1.73×10^{5}*	*6.45×10^{-1}*
RBF SVR	*2.46×10^{2}*	*1.86×10^{-1}*	***1.46×10^{5}***	***7.00×10^{-1}***
ANN	*2.44×10^{2}*	*1.86×10^{-1}*	*1.49×10^{5}*	*6.95×10^{-1}*

Note: *Italic* indicates better performance than benchmark model; **bold** indicates best performance regarding a metric.

TABLE VIII

VALIDATION PERFORMANCE FOR TOP 10% MEAN RAIN RATE BY MODELS WITH 30-MIN MSL DATASET

Model	MAE	MAPE	MSE	R^2
Benchmark	9.66×10^{2}	2.72×10^{-1}	2.49×10^{6}	7.04×10^{-1}
Lasso	*9.05×10^{2}*	***2.49×10^{-1}***	*2.08×10^{6}*	*7.53×10^{-1}*
RFR	9.72×10^{2}	*2.51×10^{-1}*	2.69×10^{6}	6.81×10^{-1}
Linear SVR	***8.97×10^{2}***	*2.50×10^{-1}*	***2.06×10^{6}***	***7.55×10^{-1}***
RBF SVR	*9.63×10^{2}*	*2.56×10^{-1}*	2.53×10^{6}	7.00×10^{-1}
ANN	*9.54×10^{2}*	*2.57×10^{-1}*	*2.25×10^{6}*	*7.32×10^{-1}*

Note: *Italic* indicates better performance than benchmark model; **bold** indicates best performance regarding a metric.

rate and top 10% mean rain rate, in comparison with merely using the output variable values at a prior time.

Regarding the predictive performances of ML regression models with different prediction time, our study showed that, consistently with intuition, the 30-min models produced significantly better validation metric values than the 60-min models. As per different ML regression modeling methods, they resulted in similar predictive performances in general. No single method was shown to be consistently superior to the others in terms of the validation metrics. However, RFR was less successful than the other methods in producing best metric values across all circumstances of model calibrations. This result implied that most of the ML regression methods were potentially capable of nowcasting rainfall. Whether one method led to a better predictive performance than another one was to a large extent stochastically determined by the specific dataset used for model calibration.

C. Classification Modeling

Validation of the ML classification models was achieved via derivation of the metrics of FPR (20), precision (21), recall (22), and F1 score (23) on the validation datasets. Tables IX and X display the model performance for input of 30 and 60 min prior, respectively. Each row corresponds to the result of an ML classification model calibrated with a specific storm cell type.

Among models for individual storm cell types of CC, MCC, SLD, and SLP, CC models resulted in the poorest performance, as manifested by the lowest F1 scores (Tables IX and X). Despite with the highest recalls, CC models yielded the lowest precisions. This was because CC storm cells were much

TABLE IX

PERFORMANCE OF 30-MIN CLASSIFICATION MODELS

Model	False Positive Rate	Precision	Recall	F1 Score
CC	0.12	0.13	0.91	0.23
MCC	0.13	0.66	0.87	0.75
SLD	0.09	0.57	0.59	0.58
SLP	0.09	0.58	0.80	0.67
MSL	0.14	0.63	0.88	0.73
All	0.15	0.50	0.93	0.65

TABLE X

PERFORMANCE OF 60-MIN CLASSIFICATION MODELS

Model	False Positive Rate	Precision	Recall	F1 Score
CC	0.21	0.08	0.80	0.14
MCC	0.22	0.55	0.79	0.65
SLD	0.12	0.41	0.60	0.49
SLP	0.13	0.45	0.67	0.54
MSL	0.23	0.53	0.82	0.65
All	0.22	0.46	0.86	0.60

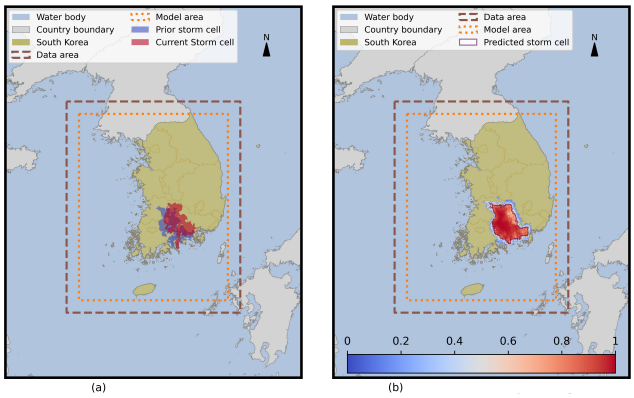


Fig. 8. Example of classification modeling for predicting location of an MCC storm cell with the MCC model at timestamp of KST 2020-09-09 07:20:00 with input data of 30 min prior. (a) Detected previous and current storm cell locations. (b) Predicted storm cell location.

smaller than the storm cells of other types. Although false positives of CC models were relatively small with respect to all pixels of an image, the true positives were also small such that the predicted area of a storm cell tended to be much larger relative to the observed area of the storm cell than in the cases of the other storm cell types.

Unlike CC models, the MCC models produced the highest precisions as well as high recalls, resulting in the best F1 scores among the models for the four individual types of storm cells for both 30-min (Table IX) and 60-min predictions (Table X). This was not only because MCC storm cells were much larger than the CC storm cells but also because there were much more data points with a better representation of the time period of data to train the MCC models than the data points for training the SLD and SLP models. Fig. 8 shows an example of the classification modeling to produce a 30-min prediction for an MCC storm cell.

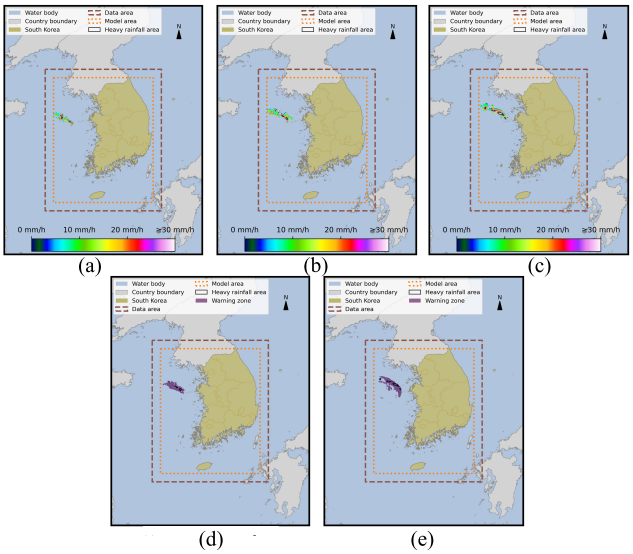


Fig. 9. Example of early warning for an SLD storm cell with the ANN model for regression modeling and SLD model for classification modeling. Detected rainfall distributions are shown for timestamps of (a) KST 2020-07-19 06:00:00, (b) KST 2020-07-19 06:30:00, and (c) KST 2020-07-19 07:00:00. (d) 30-min warning for KST 2020-07-19 06:30:00 based on data at KST 2020-07-19 06:00:00. (e) 60-min warning for KST 2020-07-19 07:00:00 based on data at KST 2020-07-19 06:00:00.

With fewer data points for training than for the MCC models, the SLD and SLP models yielded slightly lower precisions and recalls than the MCC models. In spite of this, the SLD and SLP models produced the lowest FPRs (Tables IX and X). To form larger datasets to train ML classification models for SLD and SLP storm cells, the MCC, SLD, and SLP datasets were pooled to form the MSL datasets. The MSL datasets were used to train the MSL models that resulted in even higher recalls than the MCC models while maintaining high precisions.

Meanwhile, the All datasets were created by pooling datasets of all storm cell types. With the All datasets for each prior time, only one ML classification model was trained for all individual types of storm cells. Interestingly, the all models for prior times of 30 and 60 min both yielded the highest recalls among all considered models (Tables IX and X). Although the precisions of all models were lower than the ones of MCC, SLD, SLP, and MSL models, they were much improved compared to the ones of CC models. As a result, the all models produced good F1 scores.

D. Early Warning

Based on the calibrated ML regression models for nowcasting heavy rainfall given a storm cell and ML classification models for predicting locations of storm cells, a 30- or 60-min early warning of heavy rainfall may be issued when information on the input variables becomes available for computation. Fig. 9 shows an example of early warning, based on 30- and 60-min nowcasting, with mapping of an area predicted to be covered by an SLD storm cell with potential of heavy rainfall with the application of the ANN model for assessment of variable importance and the SLD model for classification.

V. Conclusion

In this study, five ML regression modeling methods were used to assess the relative importance of radar variables derived from dual-polarimetric radar data in nowcasting 30- and 60-min heavy rainfall for the Korean Peninsula. Consistently with intuition, the 30-min models were found to provide better predictions than the 60-min models. For demonstration of their utility, the ML regression models were also coupled with ML classification models that were used to predict locations of storm cells, to provide early warnings of heavy rainfall. Regression results showed that although linear and nonlinear ML regression models may quantify the variable importance slightly differently, the rankings of important input variables determined by the regression models tended to converge in general. Radar variables of storm area averaged reflectivity, storm area volume, storm area maximum vertically integrated liquid water, echo bottom height, and longest radius were identified as highly important predictors of rainfall. Despite the commonly identified radar variables as pertinent predictors, regression results also indicated that the rankings of relative importance of radar variables were distinctive by storm cell type. This result suggests that rainfall nowcasting based on weather radar data needs to include MCS type information. As the classification results also showed different model performances among storm cell types, future work needs to focus on modeling separately for each MCS type to improve model performance in rainfall nowcasting and provision of early warnings of heavy rainfall.

Although the presented ML regression and classification models can be applied to produce early warnings of heavy rainfall, there are several issues around this application that are worth highlighting for practical purposes. First, the warning is based on the threshold of top 10% mean rain rate at 30 mm/h. To what extent this threshold can be explained as corresponding to the spectrum of hazard strength of a rainfall event at a timestamp remains unclear. Second, because the regression models for variable importance only produce predictions for an entire storm cell, the early warning is associated with the entire predicted area of the storm cell. However, the actual locations that may need an early warning may be only a fraction of the entire area of the storm cell. Third, as the precisions of the ML classification models are less than 0.7, these models still lack capabilities of providing highly reliable predictions of locations of storm cells so that there can be a high rate of false alarms for locations regarding warnings of heavy rainfall. To improve early warning modeling for heavy rainfall, future work needs to focus on resolving these issues.

References

[1] American Meteorological Society. (Jan. 26, 2012). *Nowcast*. [Online]. Available: https://glossary.ametsoc.org/wiki/Nowcast

[2] A. J. Schroeder et al., "The development of a flash flood severity index," *J. Hydrol.*, vol. 541, pp. 523–532, Oct. 2016.

[3] F. Silvestro et al., "Impact-based flash-flood forecasting system: Sensitivity to high resolution numerical weather prediction systems and soil moisture," *J. Hydrol.*, vol. 572, pp. 388–402, May 2019.

[4] J. Ritter, M. Berenguer, C. Corral, S. Park, and D. Sempere-Torres, "ReAFFIRM: Real-time assessment of flash flood impacts—A regional high-resolution method," *Environ. Int.*, vol. 136, Mar. 2020, Art. no. 105375.

[5] S. Ravuri et al., "Skilful precipitation nowcasting using deep generative models of radar," *Nature*, vol. 597, no. 7878, pp. 672–677, Sep. 2021.
[6] Z. Sokol and P. Pešice, "Comparing nowcastings of three severe convective events by statistical and NWP models," *Atmos. Res.*, vol. 93, nos. 1–3, pp. 397–407, Jul. 2009.
[7] C. J. Short and J. Petch, "Reducing the spin-up of a regional NWP system without data assimilation," *Quart. J. Roy. Meteorological Soc.*, vol. 148, no. 745, pp. 1623–1643, Apr. 2022.
[8] R. McTaggart-Cowan et al., "Modernization of atmospheric physics parameterization in Canadian NWP," *J. Adv. Model. Earth Syst.*, vol. 11, no. 11, pp. 3593–3635, Nov. 2019.
[9] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-C. Woo, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2015, pp. 1–9.
[10] X. Shi et al., "Deep learning for precipitation nowcasting: A benchmark and a new model," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2017, pp. 1–11.
[11] A. Su, H. Li, L. Cui, and Y. Chen, "A convection nowcasting method based on machine learning," *Adv. Meteorol.*, vol. 2020, Jan. 2020, Art. no. 5124274.
[12] G. Ayzel, M. Heistermann, A. Sorokin, O. Nikitin, and O. Lukyanova, "All convolutional neural networks for radar-based precipitation nowcasting," *Proc. Comput. Sci.*, vol. 150, pp. 186–192, Jan. 2019.
[13] S.-H. Moon, Y.-H. Kim, Y. H. Lee, and B.-R. Moon, "Application of machine learning to an early warning system for very short-term heavy rainfall," *J. Hydrol.*, vol. 568, pp. 1042–1054, Jan. 2019.
[14] L. Foresti, I. V. Sideris, D. Nerini, L. Beusch, and U. Germann, "Using a 10-year radar archive for nowcasting precipitation growth and decay: A probabilistic machine learning approach," *Weather Forecasting*, vol. 34, no. 5, pp. 1547–1569, Oct. 2019.
[15] M. Łoś, K. Smolak, G. Guerova, and W. Rohm, "GNSS-based machine learning storm nowcasting," *Remote Sens.*, vol. 12, no. 16, p. 2536, Aug. 2020.
[16] K. Shin, J. J. Song, W. Bang, and G. Lee, "Quantitative precipitation estimates using machine learning approaches with operational dual-polarization radar data," *Remote Sens.*, vol. 13, no. 4, p. 694, Feb. 2021.
[17] F. Zhang, X. Wang, J. Guan, M. Wu, and L. Guo, "RN-Net: A deep learning approach to 0–2 hour rainfall nowcasting based on radar and automatic weather station data," *Sensors*, vol. 21, no. 6, p. 1981, Mar. 2021.
[18] L. Han, J. Sun, W. Zhang, Y. Xiu, H. Feng, and Y. Lin, "A machine learning nowcasting method based on real-time reanalysis data," *J. Geophys. Res., Atmos.*, vol. 122, no. 7, pp. 4038–4051, Apr. 2017.
[19] G. Franch, D. Nerini, M. Pendesini, L. Coviello, G. Jurman, and C. Furlanello, "Precipitation nowcasting with orographic enhanced stacked generalization: Improving deep learning predictions on extreme events," *Atmosphere*, vol. 11, no. 3, p. 267, Mar. 2020.
[20] V. Bouget, D. Béréziat, J. Brajard, A. Charantonis, and A. Filoche, "Fusion of rain radar images and wind forecasts in a deep learning model applied to rain nowcasting," *Remote Sens.*, vol. 13, no. 2, p. 246, Jan. 2021.
[21] M. Dixon and G. Wiener, "TITAN: Thunderstorm identification, tracking, analysis, and nowcasting—A radar-based methodology," *J. Atmos. Ocean. Technol.*, vol. 10, no. 6, pp. 785–797, Dec. 1993.
[22] J. T. Johnson et al., "The storm cell identification and tracking algorithm: An enhanced WSR-88 D algorithm," *Weather Forecasting*, vol. 13, no. 2, pp. 263–276, Jun. 1998.
[23] P. J. Rossi, V. Chandrasekar, V. Hasu, and D. Moisseev, "Kalman filtering–based probabilistic nowcasting of object-oriented tracked convective storms," *J. Atmos. Ocean. Technol.*, vol. 32, no. 3, pp. 461–477, Mar. 2015.
[24] R. Kato, S. Shimizu, K. Shimose, T. Maesaka, K. Iwanami, and H. Nakagaki, "Predictability of meso-γ-scale, localized, extreme heavy rainfall during the warm season in Japan using high-resolution precipitation nowcasts," *Quart. J. Roy. Meteorological Soc.*, vol. 143, no. 704, pp. 1406–1420, Apr. 2017.
[25] R. J. Doviak and D. S. Zrnić, *Doppler Radar and Weather Observations*, 2nd ed. San Diego, CA, USA: Academic, 1993.
[26] G. W. Lee, "Sources of errors in rainfall measurements by polarimetric radar: Variability of drop size distributions, observational noise, and variation of relationships between R and polarimetric parameters," *J. Atmos. Ocean. Technol.*, vol. 23, no. 8, pp. 1005–1028, Aug. 2006.
[27] Y. Wang and V. Chandrasekar, "Quantitative precipitation estimation in the CASA X-band dual-polarization radar network," *J. Atmos. Ocean. Technol.*, vol. 27, no. 10, pp. 1665–1676, Oct. 2010.
[28] National Weather Service. (Mar. 31, 2022). *What Is Dual-Pol?* [Online]. Available: https://www.weather.gov/jan/dualpolupgrade-about
[29] S.-H. Jung, "Radar-based tracking and statistical characteristics of convective storm activities," Ph.D. dissertation, Dept. Astron. Atmos. Sci., Kyungpook National Univ., Daegu, South Korea, 2013.
[30] Global Administrative Areas (GADM). (Sep. 10, 2021). *Download GADM Data (Version 2.8)*. [Online]. Available: https://gadm.org/download_country_v2.html
[31] J. H. Lambert and W. R. Tobler, *Notes and Comments on the Composition of Terrestrial and Celestial Maps*. Redlands, CA, USA: Esri Press, 2011.
[32] S.-H. Jung and G. Lee, "Radar-based cell tracking with fuzzy logic approach," *Meteorologic Appl.*, vol. 22, no. 4, pp. 716–730, Oct. 2015.
[33] S. Kwon, S.-H. Jung, and G. Lee, "Inter-comparison of radar rainfall rate using constant altitude plan position indicator and hybrid surface rainfall maps," *J. Hydrol.*, vol. 531, pp. 234–247, Dec. 2015.
[34] T.-Y. Lee and Y.-H. Kim, "Heavy precipitation systems over the Korean Peninsula and their classification," *Asia–Pacific J. Atmos. Sci.*, vol. 43, pp. 367–396, Nov. 2007.
[35] QGIS Project. (Oct. 29, 2021). *QGIS: A Free and Open Source Geographic Information System*. [Online]. Available: https://qgis.org/en/site/
[36] United States Geological Survey (USGS). (Sep. 10, 2021). *GMTED2010 Viewer*. [Online]. Available: https://topotools.cr.usgs.gov/gmted_viewer/viewer.htm
[37] Python Software Foundation. (Sep. 10, 2021). *Python Software Foundation*. [Online]. Available: https://www.python.org/psf/
[38] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. Roy. Stat. Soc., B, Methodol.*, vol. 58, no. 1, pp. 267–288, Jan. 1996.
[39] Y. V. Wang, P. Gardoni, C. Murphy, and S. Guerrier, "Empirical predictive modeling approach to quantifying social vulnerability to natural hazards," *Ann. Amer. Assoc. Geographers*, vol. 111, no. 5, pp. 1559–1583, Jul. 2021.
[40] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, pp. 5–32, Oct. 2001.
[41] P. Geurts, D. Ernst, and L. Wehenkel, "Extremely randomized trees," *Mach. Learn.*, vol. 63, no. 1, pp. 3–42, Mar. 2006.
[42] A. J. Smola and B. Schölkopf, "A tutorial on support vector regression," *Statist. Comput.*, vol. 14, no. 3, pp. 199–222, Aug. 2004.
[43] F. Rosenblatt, "The perceptron: A probabilistic model for information storage and organization in the brain," *Psychol. Rev.*, vol. 65, no. 6, pp. 386–408, 1958.
[44] Scikit-Learn. (Mar. 31, 2022). *Scikit-Learn: Machine Learning in Python*. [Online]. Available: https://scikit-learn.org/stable/
[45] K. Fukushima, "Visual feature extraction by a multilayered network of analog threshold elements," *IEEE Trans. Syst. Sci. Cybern.*, vol. SSC-5, no. 4, pp. 322–333, Oct. 1969.
[46] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," *Proc. Mach. Learn. Res.*, vol. 15, pp. 315–323, Jun. 2011.
[47] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," presented at the 3rd Int. Conf. Learn. Represent., San Diego, CA, USA, May 2015.
[48] TensorFlow. (Sep. 10, 2021). *An End-to-End Open Source Machine Learning Platform*. [Online]. Available: https://www.tensorflow.org/
[49] A. Altmann, L. Toloşi, O. Sander, and T. Lengauer, "Permutation importance: A corrected feature importance measure," *Bioinformatics*, vol. 26, pp. 1340–1347, May 2010.
[50] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.
[51] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (ELUs)," presented at the 4th Int. Conf. Learn. Represent., San Juan, Puerto Rico, May 2016.

Yi Victor Wang received the Ph.D. degree in civil engineering with the direction of societal risk management from the University of Illinois at Urbana–Champaign, Champaign, IL, USA, in 2018.

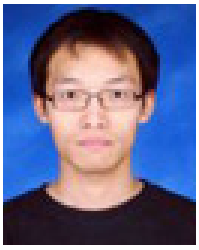
He is currently serving as a Postdoctoral Fellow with the Institute for Earth, Computing, Human and Observing (ECHO), Chapman University, Orange, CA, USA. His research interests include hazard modeling, machine learning, disaster vulnerability, risk analysis, and media coverage of crises.

Dr. Wang is a member of the American Geophysical Union, the American Society of Civil Engineers, the Earthquake Engineering Research Institute, and the European Geosciences Union.

Seung Hee Kim (Member, IEEE) received the Ph.D. degree in atmospheric and oceanic sciences from the University of California at Los Angeles, Los Angeles, CA, USA, in 2010.

His research interests include dynamics and physics of severe weather and mesoscale convective system, and numerical weather prediction.

Dr. Kim is a member of the American Geophysical Union and the Asia-Oceania Geoscience Society.



Geunsu Lyu received the master's degree in astronomy and atmospheric sciences from Kyungpook National University, Daegu, South Korea, in 2015.

He has conducted research on storm identifying and tracking, precipitation estimation and forecasting using dual polarization weather radars. Since 2012, he has been a Researcher with the Kyungpook National University's Center for Atmospheric Remote Sensing Research (CARE), Daegu. His research interests include radar meteorology and hydrology and radar signal processing.

Choeng-Lyong Lee received the master's degree from the Department of Astronomy and Atmospheric Sciences, Kyungpook National University, Daegu, South Korea, in 2015.

He is currently serving as a Researcher with the Center for Atmospheric Remote Sensing Research (CARE), Kyungpook National University. His research interests include radar meteorology, microphysics, and dynamical analysis of clouds system using radar and satellite observations.

Mr. Lee is a member of the Korean Meteorological Society.

Gyuwon Lee received the Ph.D. degree in atmospheric sciences with focus on radar meteorology and precipitation physics from McGill University, Montreal, QC, Canada, in 2003.

He is a Professor of atmospheric sciences with the School of Earth System Sciences and the Director of Center for Atmospheric Remote Sensing Research (CARE), Kyungpook National University (KNU), Daegu, South Korea. He was a Research Scientist with the National Center for Atmospheric Research (NCAR) and was a Radar Operator with the J. S. Marshall Radar Observatory. His research areas include quantitative precipitation estimation (QPE) and forecasting (QPF), and precipitation microphysics.

Dr. Lee received the Presidential award from the Korean government and the Mooksan award from the Korean Meteorological Society for his service to the community. He is currently a member of WMO working group on nowcasting and mesoscale research (NMR) and ISO/WMO committee on weather radar and wind profiler.

Ki-Hong Min received the Ph.D. degree in atmospheric sciences with focus on mesoscale and regional climate modeling from Purdue University, West Lafayette, IN, USA, in 2005.

He is a Professor of meteorology with the School of Earth System Sciences, Kyungpook National University, Daegu, South Korea. His research area is in mesoscale modeling, remote sensing, and weather analysis of the atmosphere. He is currently interested in high-resolution modeling study of convective storms with radar data assimilation.

Prof. Min is a member of the American Geophysical Union, the American Meteorological Society, the Asia-Oceania Geoscience Society, and the Korean Meteorological Society, where he serves as a member of the board.

Menas C. Kafatos (Life Member, IEEE) received the Ph.D. degree in physics from the Massachusetts Institute of Technology, Cambridge, MA, USA, in 1972.

He is The Fletcher Jones Endowed Professor of computational physics with Chapman University, Orange, CA, USA, and the Director of the Institute for Earth, Computing, Human and Observing (Institute for ECHO) as well as the Founding Dean of Schmid College of Science and Technology from 2009 to 2012. He is an author, a physicist, and a philosopher. He works on the environment, climate change and its effects, and natural hazards, including wildfires, pollution, storms, and droughts. He is expert on quantum mechanics, cosmology, measurement theory, and the role of the mind. He is one of few geo scientists who has worked on both Earth and space/astronomy, as well as data information systems for both Earth and space science.

Prof. Kafatos is a member of the American Geophysical Union (AGU), International Astronomical Union (IAU), American Association for the Advancement of Science (AAAS), and American Physical Society (APS). He is an Honorary Member of the Romanian Academy of Sciences elected in 2000, and a Foreign Member of the Korean Academy of Science and Technology (KAST) elected in 2018.