8-20-2023

Verifying Empirical Predictive Modeling of Societal Vulnerability to Hazardous Events: A Monte Carlo Experimental Approach

Yi Victor Wang

Seung Hee Kim

Menas C. Kafatos

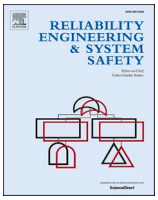
Verifying Empirical Predictive Modeling of Societal Vulnerability to Hazardous Events: A Monte Carlo Experimental Approach

Reliability Engineering and System Safety 240 (2023) 109593

Contents lists available at ScienceDirect

Reliability Engineering and System Safety

journal homepage: www.elsevier.com/locate/ress



Verifying empirical predictive modeling of societal vulnerability to hazardous events: A Monte Carlo experimental approach

Yi Victor Wang [a,b,*], Seung Hee Kim [b], Menas C. Kafatos [b]

[a] Department of Emergency Management, Massachusetts Maritime Academy, Buzzards Bay, MA, USA
[b] Institute for Earth, Computing, Human and Observing (ECHO), Chapman University, Orange, CA, USA

ARTICLE INFO

Keywords:
Hazard loss
Machine learning
Simulation
Social vulnerability
Societal system

ABSTRACT

With the emergence of large amounts of historical records on adverse impacts of hazardous events, empirical predictive modeling has been revived as a foundational paradigm for quantifying disaster vulnerability of societal systems. This paradigm models societal vulnerability to hazardous events as a vulnerability curve indicating an expected loss rate of a societal system with respect to a possible spectrum of intensity measure (IM) of an event. Although the empirical predictive models (EPMs) of societal vulnerability are calibrated on historical data, they should not be experimentally tested with data derived from field experiments on any societal system. Alternatively, in this paper, we propose a Monte Carlo simulation-based approach to experimentally test EPMs of societal vulnerability. Our study applied an eigenvalue-based method to generate data on societal experiences of IM and pre-event vulnerability indicators. True models were designed to simulate event loss data. Supervised machine learning (ML) models were then trained on simulated data and were found to provide similar predictive performances as the true models. Our results suggested that the calibrated ML-EPMs could effectively quantify societal vulnerability given a normally experienced IM. To extrapolate a vulnerability curve for large IMs, however, simple models should be preferred.

1. Introduction

To reduce disaster losses to human societal systems due to hazardous events, it is essential to accurately identify vulnerable communities and measure their societal vulnerability to hazard losses. In this article, societal vulnerability refers to the inability of a human community to withstand the intense forces of a hazardous event and the susceptibility of the community to losses due to the event. For measuring societal vulnerability, a commonly adopted method is to derive a vulnerability index, such as the social vulnerability index (SVI), as a function of pre-event indicators of societal characteristics (see, e.g., [1–8]). Without access to a sufficiently large number of historical records on intensity measures (IMs), such as peak ground acceleration (PGA) and peak sustained wind speed, and hazard losses, such as casualty and economic damage, modelers may select societal indicators based on knowledge generalized from previous case studies and expert opinions to create vulnerability indices as a convenient tool to swiftly map the spatial distribution of societal vulnerability to guide practices of hazard management. However, because of its lack of direct association with historical data on event IMs and losses, this index-based approach to quantifying societal vulnerability can hardly be satisfactorily verified in terms of whether the developed indices can be used to measure what they are purported to indicate [9–11].

Unlike modelers using the index-based approach for social vulnerability, engineering scholars have proposed empirical models to quantify system vulnerability with historical records or experimental data on IMs of hazardous conditions and the corresponding system losses (see, e.g., [12–19]). These empirical vulnerability models can be used to estimate the probability of a system being in a damage state given IMs (see, e.g., [13,20]) or to predict the expected rate of loss with respect to a unit value of system exposed to the IMs (see, e.g., [12,16]). Such empirical methods have been widely and effectively applied to quantify vulnerability of structural and infrastructural systems, such as bridges [14,18,21,22], buildings [16,19], chemical processing facilities [23], concrete civil defense structures [24], dike systems [25], industrial horizontal vessels [26], industrial process pipelines [27], natural gas pipelines [28], nuclear containment structures [29], nuclear power plants [30,31], offshore wind turbine support structures [15], power transmission systems [32–34], railway networks [35], reinforced concrete columns [36], solar panels [37], and storage tanks [38,39]. Because the empirical

* Corresponding author at: 101 Academy Drive, Kurz Hall 208, Buzzards Bay, MA 02532, USA.
E-mail address: vwang@maritime.edu (Y.V. Wang).

https://doi.org/10.1016/j.ress.2023.109593
Received 4 October 2022; Received in revised form 31 July 2023; Accepted 18 August 2023
Available online 20 August 2023

vulnerability models can provide predictions of expected probability or rate of loss of a considered system, they can be used to facilitate quantitative risk assessment to stochastically compute the expected losses to the system or losses within a broader societal context given future or simulated hazardous events (see, e.g., [40,41]). In addition, with data on IMs and system damages produced from lab or field experiments, many of these empirical models can be tested such that their predictive performances can be compared with the ones of the true models.

Given its advantages, the empirical predictive modeling approach has recently been applied to quantify societal vulnerability with consideration of societal characteristics of communities that are potentially exposed to future hazardous events (see, e.g., Wang et al. [11,42,43]). With this approach, societal vulnerability can be modeled as the relationship between the indicators of societal characteristics and the expected loss rate with respect to a unit value, such as casualty rate and economic loss rate per 1 US dollar of gross domestic product, given a spectrum of IM values. To estimate societal vulnerability in terms of the expected loss rate, quantitative modeling methods such as machine learning (ML) can be implemented based on model calibrations with historical data on IMs and event losses (see, e.g., [11,42–47]). Coupled with hazard maps showing the geographic distributions of locational probabilities of exceeding threshold IMs within a considered period, the predicted societal vulnerability in terms of the expectation of a loss rate can be applied to quantitatively assess and geographically map disaster risks in terms of the expected loss due to future hazardous events. As such a risk assessment can be conducted with regards to a specific type of loss, this computational application of the empirical predictive models (EPMs) of societal vulnerability is far beyond the utility of the index-based approach that merely produces an abstract index to quantify societal vulnerability. Despite the advantages of the EPMs, however, it is unethical to conduct a purely experimental study to verify the EPMs of societal vulnerability with real-world data, as it should be forbidden to generate data points on IMs and hazard losses with experiments on human communities.

To experimentally verify the EPMs of societal vulnerability, in this paper, we propose a Monte Carlo (MC) approach to simulate IMs, indicators of societal vulnerability, and event losses based on historical records to train EPMs and compare the performances of the trained EPMs and predetermined true models. Given the usual large numbers of data points with zero losses in hazardous events, a 2-part model is ideal for prediction of loss rate (see, e.g., [42–44,48]). Accordingly, a true model in this study was comprised of 2 parts. The first part, or the true classification model, was a logistic regression (LR) model that simulated whether a data point, corresponding to a community that had experienced an IM of a hazardous event, involved positive loss. The second part, or the true regression model, was a multiple linear regression (MLR) model that simulated the loss rate of the community given positive loss. Two true models were created for 2 scenarios. For testing model calibrations, the first scenario involved 1 IM and 1 vulnerability indicator (VI) for both classification and regression modeling. For testing variable selections, the second scenario involved 1 IM and 2 VIs in both parts of the model. The examined EPMs included 8 supervised ML models, 4 for classification and the other 4 for regression, corresponding to the first and second parts of the true model, respectively. For each scenario, 16 joint models were derived. The ML-EPMs were trained on 10,000 simulated data points and tested on another 10,000 simulated data points. Data simulation and model verification were achieved via open-source programming language Python 3.8.11 [49].

The academic significance of the present study is manifested in three aspects. First, the study is the first scholarly effort to develop an MC method to verify, stochastically and experimentally, EPMs of societal vulnerability to hazardous events. Second, for data simulation, we adopted a novel eigenvalue-based approach to empirically mimic the real-world correlational structure of variables of IM and VIs. Third, we propose a permutation importance-based algorithm to select pertinent input variables of EPMs, significantly improving the computational efficiency compared to the traditional approaches to selecting VIs to model vulnerability.

The rest of the paper has 4 sections. Section 2 introduces the methods for data simulation, including a preliminary analysis of historical data on IMs and losses of hazardous events, simulation of input variables of EPMs, and simulation of event losses based on the true models. Section 3 covers the process of model verification, including the formulation of the joint models for estimating societal vulnerability, calibration of ML-EPMs, selection of input variables of EPMs, and validation of the trained EPMs. Section 4 showcases, with appropriate discussions, the results of the study, including predictive performances of calibrated models, generation of vulnerability curves, and analysis of sensitivity of EPMs to different numbers of training data points (TDPs). Section 5 concludes the paper with summaries of the study and suggestions for future work.

2. DATA simulation

In this study, two sets of data on IM and VIs were generated for two scenarios for testing model calibrations and variable selections, respectively. The first scenario (1V) assumed that the input of the true model only included 1 IM and 1 VI. In Scenario 1V, the EPMs were trained accordingly on data with these two input variables only. The second scenario (nV) simulated 1 IM and 10 VIs but assumed that the loss rate was determined by the IM and 2 VIs only. In Scenario nV, the EPMs were trained on data with 10 VIs to identify the pertinent input variables. The adoption of 10 candidate and 2 pertinent VIs for scenario nV in the study was performed for a simple and straightforward demonstration. For each scenario, a true model was designed to produce a vulnerability curve that was consistent with expert intuition and similar to the ones empirically derived from previous studies.

2.1. Preliminary data analysis

To take into consideration the correlational structures of the input variables for data simulation, we first examined the correlations between input variables of historical data collected from 4 previous studies, including WGMG19 [42], WGMG20 [43], WGMG21 [11], and WS21 [48]. Among these studies, models of WGMG19, WGMG20, and WGMG21 were for earthquakes in Taiwan, the world, and Nepal, respectively, using PGA as the sole IM. Meanwhile, WS21 was for floods due to hurricanes in North Carolina, with mean water depth as the only IM for its model. Apart from WGMG20 which only included the data of World Development Indicators (WDI) from the World Bank [50] for constructing VIs, all the other studies adopted the local census data as the main source for data on VIs. All the data on input variables from the previous studies were transformed, standardized, and normalized, when necessary, such that all input variables of IMs and VIs in this study had a mean of 0 and standard deviation of 1. Such a data processing was necessary for keeping the examined historical data consistent with the simulated data proper for training the ML-EPMs.

With the standard-normalization of input data, we examined the correlations between the standard-normalized input variables. Our examination suggested that there were correlations between IM and VIs (I–V correlations), as shown in Fig. 1(a)–(d). Such I–V correlations may be associated with the spatial autocorrelations of the data points. In particular, the absolute I–V correlations could even reach about 0.8 in WGMG21 (Fig. 1(c)). This may be because the data for WGMG21 were collected based on one single earthquake event for one country with a relatively small area. Regarding the correlations between VIs (V–V correlations), as shown in Fig. 1(e)–(h), the census-based data tended to have a bell-shaped distribution of V–V correlations (Fig. 1(e), (g), and (h)). However, many of the VIs had large absolute correlations with each other (Fig. 1(f)) in the case of WGMG20, which was based on the WDI data of the World Bank [50].

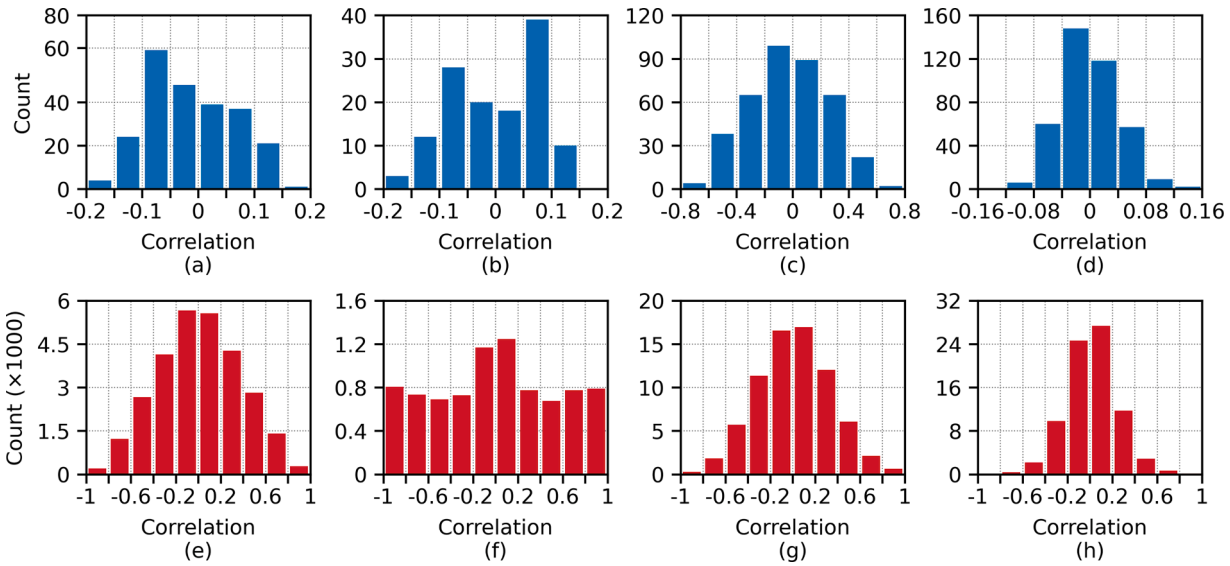


Fig. 1. Distributions of correlations between a sole intensity measure and vulnerability indicators (I–V) and between vulnerability indicators (V–V) based on empirical data collected in previous studies: (a) I–V from WGMG19, (b) I–V from WGMG20, (c) I–V from WGMG21, (d) I–V from WS21, (e) V–V from WGMG19, (f) V–V from WGMG20, (g) V–V from WGMG21, and (h) V–V from WS21.

2.2. Simulation of input variables

As the distributions of IM and vulnerability variables were to a large extent similar across all 4 examined studies, we focused on using the WS21 data for demonstration of our approach in this study. To consider the correlational structures of input variables of true models and EPMs, we applied an eigenvalue-based MC simulation method.

2.2.1. Scenario of one vulnerability indicator

In Scenario 1V, there were only 1 IM and 1 VI. Accordingly, no V–V correlations were involved. In this scenario, the I–V correlation matrix was 2 by 2 and only had 1 free eigenvalue. As shown in Fig. 2(a), we modeled this free eigenvalue with a beta distribution:

$$f_{\text{Beta}}(x_{\text{Beta}}, a_{\text{Beta}}, b_{\text{Beta}}) = \frac{\Gamma(a_{\text{Beta}} + b_{\text{Beta}}) x_{\text{Beta}}^{a_{\text{Beta}}-1} (1 - x_{\text{Beta}})^{b_{\text{Beta}}-1}}{\Gamma(a_{\text{Beta}})\Gamma(b_{\text{Beta}})}, \tag{1}$$

where $x_{\text{Beta}} \in [0, 1]$ was the random variable of the free eigenvalue, $\Gamma(\cdot)$ was the gamma function, and $a_{\text{Beta}} > 0$ and $b_{\text{Beta}} > 0$ were 2 model parameters. With the empirical beta model, we simulated the free eigenvalue. The correlation matrix was then simulated based on the simulated free eigenvalue. Next, the I–V correlation matrix was used to further simulate the values of the IM and VI for 20,000 data points, with the assumption that the IM and VI followed a bivariate normal distribution. This distribution could be written in a generalized form for multivariate normal distribution:

$$f_{\text{MN}}(\boldsymbol{x}_{\text{MN}}, \boldsymbol{\mu}_{\text{MN}}, \boldsymbol{\Sigma}_{\text{MN}}) = \det(2\pi\boldsymbol{\Sigma}_{\text{MN}})^{-\frac{1}{2}} \exp\left[-\frac{1}{2}(\boldsymbol{x}_{\text{MN}} - \boldsymbol{\mu}_{\text{MN}})^{\text{T}} \boldsymbol{\Sigma}_{\text{MN}}^{-1} (\boldsymbol{x}_{\text{MN}} - \boldsymbol{\mu}_{\text{MN}})\right], \tag{2}$$

where $\boldsymbol{x}_{\text{MN}}$ was the column vector of random input variables, $\boldsymbol{\mu}_{\text{MN}}$ was the column vector of locational parameters, $\boldsymbol{\Sigma}_{\text{MN}}$ was the covariance matrix of $\boldsymbol{x}_{\text{MN}}$, and $\det(\cdot)$ was the determinant function. Because the input data were standard-normalized, the locational parameters of $\boldsymbol{\mu}_{\text{MN}}$ equaled 0 and $\boldsymbol{\Sigma}_{\text{MN}}$ was the same as the correlation matrix of input variables. A repeat of this simulation process produced the distribution of I–V correlations as shown in Fig. 2(b).

2.2.2. Scenario of multiple vulnerability indicators

In Scenario nV, the input variables for EPMs included 1 IM and 10 VIs. For this scenario, based on a set of eigenvalues of the correlation matrix of input variables, we chose to identify 1 simulated input variable as the IM and treat the other 10 as the VIs after the simulation of input variables. Accordingly, the correlation matrix of these input variables had the size of 11 by 11, corresponding to 11 eigenvalues including 10

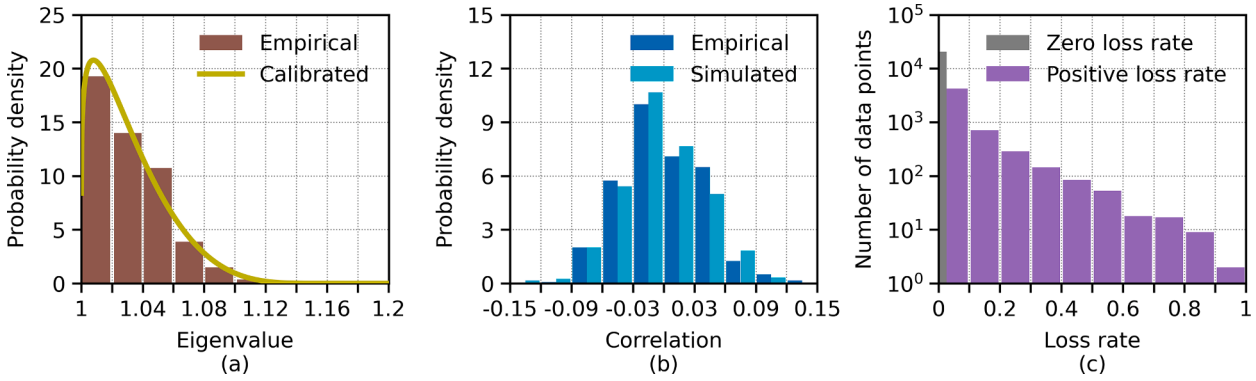


Fig. 2. Data simulation for Scenario 1V based on data of WS21: (a) empirically derived distribution of the free eigenvalue of correlation matrix between intensity measure and one vulnerability indicator; (b) empirical and simulated distributions of correlation between intensity measure and one vulnerability indicator; and (c) distribution of number of simulated data points of societal vulnerability by the true model.

free eigenvalues (E0–E9). We modeled these 10 free eigenvalues with a multivariate normal distribution (Eq. (2)) with the model parameters empirically derived from the historical data, as shown in Fig. 3(a). We then simulated these 10 free eigenvalues and used the simulated eigenvalues to simulate a correlation matrix of 11 input variables. For each simulated input variable, we derived its correlations with the other simulated variables and conducted a Kolmogorov-Smirnov (KS) test [51] on the distribution of these correlations compared to the empirical I–V correlations of the historical data. A threshold of KS test p-value was set at 0.9 such that if none of the simulated correlations was greater than the threshold, we repeated the simulation process until the threshold was reached. A large threshold of 0.9 was used because the KS test had an inverse burden of proof for rejection of null hypothesis. Next, we selected the simulated variable with the largest KS test p-value as the IM and treated the other simulated variables as the VIs. Fig. 3(b) and (c) show the empirical cumulative distributions of the I–V and V–V correlations, respectively, with comparisons between the observations and simulations.

2.3. Simulation of loss rates

For either scenario, we performed 1 simulation of data on input variables to generate 20,000 data points. We then applied a true model to produce data on loss rates. A loss rate was derived with 2 steps. For an ith data point, the first step was to compute the expected probability of experiencing positive loss given IM and VIs with an LR model

$$y_{1i} = \frac{\exp(\boldsymbol{X}_i\boldsymbol{\beta}_{\mathrm{LR}})}{\exp(\boldsymbol{X}_i\boldsymbol{\beta}_{\mathrm{LR}}) + 1}, \tag{3}$$

where y_{1i} was the expected probability, $\boldsymbol{X}_i$ was the ith simulation of input data matrix $\boldsymbol{X}$, and $\boldsymbol{\beta}_{\mathrm{LR}}$ was the column vector of model coefficients. We then simulated whether the data point contained a zero or positive loss rate based on y_{1i}. If the data point had a positive loss, the second step simulated the value of the positive loss rate with an inverse logistic transformation of prediction of an MLR model described by

$$y_{2i} = \frac{\exp(y_{\mathrm{R}i})}{\exp(y_{\mathrm{R}i}) + 1} \tag{4}$$

and

$$y_{\mathrm{R}i} = \boldsymbol{X}_i\boldsymbol{\beta}_{\mathrm{MLR}} + \sigma_{\mathrm{MLR}}\varepsilon_{\mathrm{MLR}}, \tag{5}$$

where y_{2i} was the simulated loss rate, $y_{\mathrm{R}i}$ was the output of model prediction, $\boldsymbol{\beta}_{\mathrm{MLR}}$ was the column vector of model coefficients, σ_{MLR} was the dispersion parameter, and $\varepsilon_{\mathrm{MLR}}$ was an independent standard normal random variable. The distributions of simulated loss rates are displayed in Figs. 2(c) and 3(d) for Scenarios 1V and nV, respectively.

For Scenario 1V, $\boldsymbol{X}$ had 3 columns, corresponding to the intercept, IM, and VI. Accordingly, $\boldsymbol{\beta}_{\mathrm{LR}}$ and $\boldsymbol{\beta}_{\mathrm{MLR}}$ were set with values shown in Table 1. For Scenario nV, $\boldsymbol{X}$ for the true model had 4 columns, corresponding to the intercept, IM, and 2 VIs. Among the 10 VIs (V0–V9) simulated for Scenario nV, only the first 2 VIs (V0 and V1) were used for prediction with the true model. Table 2 lists the values of $\boldsymbol{\beta}_{\mathrm{LR}}$ and $\boldsymbol{\beta}_{\mathrm{MLR}}$ for Scenario nV. For both scenarios, σ_{MLR} was set at 1.2 to add a desirable amount of uncertainty to the simulation of loss rate. Before training EPMs, we randomly split the dataset of either scenario into a training and testing datasets, each with 10,000 data points.

Table 1
Parameters of true models for Scenario 1V.

Model coefficients	Intercept	IM	VI
$\boldsymbol{\beta}_{\mathrm{LR}}$	–1.2	1	0.5
$\boldsymbol{\beta}_{\mathrm{MLR}}$	–4	1	0.5

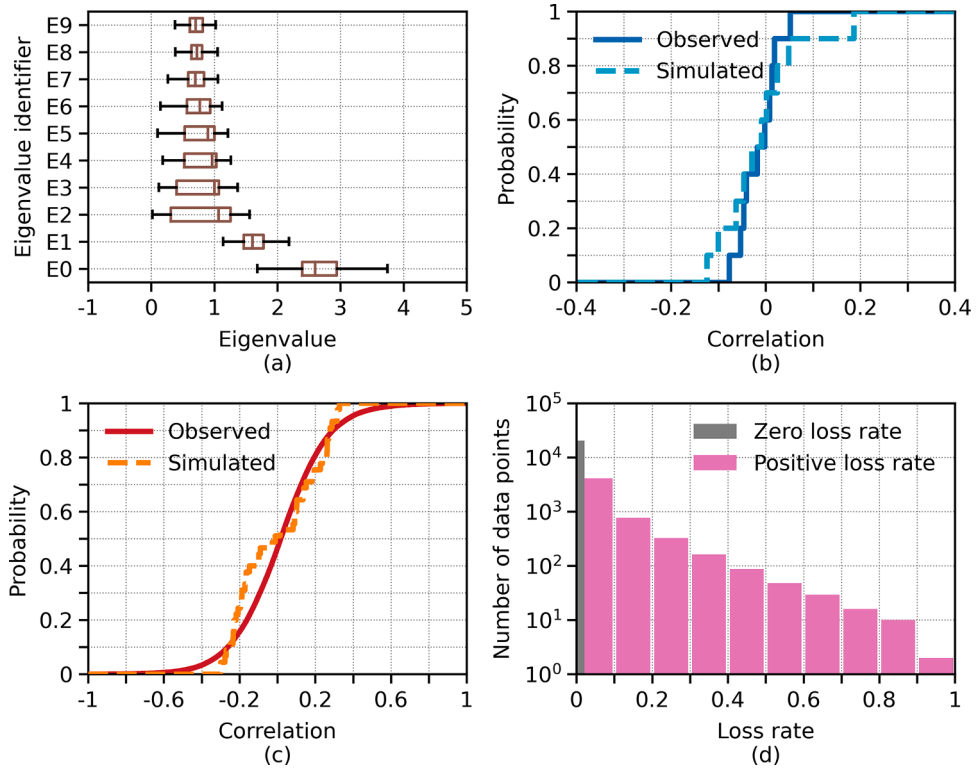


Fig. 3. Data simulation for Scenario nV based on data of WS21: (a) empirically derived distribution of the free eigenvalues of correlation matrix between 11 variables including the IM and randomly selected 10 VIs; (b) observed and simulated distribution of correlations between IM and 1 VI; (c) observed and simulated distribution of correlations between VIs; and (d) distribution of number of simulated data points of societal vulnerability by the true model.

Table 2
Parameters of true models for Scenario nV.

Model coefficients	Intercept	IM	V0	V1
$\boldsymbol{\beta}_{LR}$	–1.2	1	0.5	0.4
$\boldsymbol{\beta}_{MLR}$	–4	1	0.5	0.4

3. Model verification

Consistently with the historical data on hazard loss, a large number of data points were simulated to contain zero losses in this study. To accommodate the issue of inflation of zero data points, a 2-part modeling approach was used, as recommended by previous works on modeling loss rate (see, e.g., [42–44,48]). This adopted approach was also consistent with the 2-step simulation of loss rate with the true models. For either Scenario 1V or nV, the first part of the modeling process provided the expected probability of a data point containing positive loss. Given positive loss, the second part computed the conditional expected positive loss rate of the data point. A joint model then integrated the results of the 2 parts to predict the expected loss rate as the measure of societal vulnerability. For verification of this modeling process, we tested 8 supervised ML models, with 4 classification and 4 regression models corresponding to the first and second parts, respectively.

3.1. Joint model formulation

For either scenario in this study, 16 joint calibrated models were tested against the joint true model. The joint calibrated models were the combination of 4 classification models and 4 regression models. A joint calibrated model predicted the societal vulnerability in terms of expected loss rate of a data point

$$\widehat{y}_{JMi} = \widehat{y}_1(\boldsymbol{X}_i, \widehat{\boldsymbol{\theta}}_1)\widehat{y}_2(\boldsymbol{X}_i, \widehat{\boldsymbol{\theta}}_2), \tag{6}$$

where $\widehat{y}_{JMi}$ was the ith estimated output of the joint model, $\widehat{y}_1(\cdot)$ and $\widehat{y}_2(\cdot)$ were respectively the estimates of the classification model of the first part and the regression model of the second part, and $\widehat{\boldsymbol{\theta}}_1$ and $\widehat{\boldsymbol{\theta}}_2$ were the calibrated model parameters.

3.2. Classification models

In this study, we applied 4 supervised classification modeling methods to estimate the expected probability of a data point containing positive loss given the input of IM along with 1 VI for Scenario 1V or multiple VIs for Scenario nV. These classification modeling methods included LR, random forest classification (RFC), support vector classification (SVC), and artificial neural network classification (ANNC).

3.2.1. Logistic regression

LR is the most basic ML modeling method for binary classification. It can also be categorized as a statistical approach. An LR model is a generalized linear model for a Bernoulli random variable with a logit link function for the ith data point

$$\boldsymbol{X}_i\boldsymbol{\beta}_{LR} = \ln\left(\frac{y_{1i}}{1-y_{1i}}\right), \tag{7}$$

which is the inverse of Eq. (3). In the study, the LR modeling was carried out with the *LogisticRegression* function of the linear_model package of the scikit-learn library [52]. The first part of the true model for either scenario was also designed as an LR model (Tables 1 and 2).

3.2.2. Random forest classification

RFC is the classification version of the random forest (RF) modeling approach. The RF approach randomly generates a number of decision trees based on sampling of TDPs and aggregates the individual outputs of the trees to provide an ensemble output for prediction. To achieve RFCs in this study, we used the *RandomForestClassifier* function of the ensemble package of the scikit-learn library [52]. Accordingly, the expected probability of positive loss given by an RFC model was computed as the mean of the probabilities predicted by the trees of the RF. After trials on the training data, we adopted the entropy function

$$Entropy = -\sum_{i=1}^{n} \widehat{y}_{1i}\ln(\widehat{y}_{1i}) \tag{8}$$

as the optimal criterion for measuring the quality of split in the trees for RFC, where $\widehat{y}_{1i}$ was the predicted probability.

3.2.3. Support vector classification

SVC is a support vector machine (SVM) modeling approach for classification. An SVC model is a hyperplane in the original or transformed vector space of data points involving only the input variables. The SVC hyperplane lies at the center between 2 parallel hyperplanes sandwiching a maximized margin that separates the data points according to the 2 classes of the output variable. The data points on or near the maximum-margin hyperplanes are called the support vectors. For SVC, the TDPs can be denoted as $(\boldsymbol{x}_{SVM1}, y_{SVC1})$, $(\boldsymbol{x}_{SVM2}, y_{SVC2})$, …, $(\boldsymbol{x}_{SVMn}, y_{SVCn})$, where n is the number of TDPs, $\boldsymbol{x}_{SVMi} \in \mathscr{X}$, $\mathscr{X}$ is the vector space of input data with m dimensions, and y_{SVCi} is the output that is set to equal either 1 or –1. A hyperplane in $\mathscr{X}$ is

$$\boldsymbol{w}_{SVC}^{T}\boldsymbol{x}_{SVM} + b_{SVC} = 0, \tag{9}$$

where $\boldsymbol{w}_{SVC} \in \mathscr{X}$ is the column vector of weight parameters and $b_{SVC} \in \mathbb{R}$ is the bias parameter. The SVC hyperplane can be found by minimizing

$$L_{SVC} = \frac{1}{2}\|\boldsymbol{w}_{SVC}\|^2 + C_{SVC}\sum_{i=1}^{n}\xi_{SVCi}, \tag{10}$$

subject to

$$\begin{cases} y_{SVCi}\left(\boldsymbol{w}_{SVC}^{T}\boldsymbol{x}_{SVMi} - b_{SVC}\right) \geq 1 - \xi_{SVCi} \\ \xi_{SVCi} \geq 0 \end{cases}, \tag{11}$$

where $\|\cdot\|$ is the Euclidean norm operator, C_{SVC} is a regularization hyperparameter determining the error tolerance, and ξ_{SVCi} is the ith value of a slack variable that allows the corresponding data point to exist inside the maximized margin.

To solve the optimization problem of Eqs. (10)–(11), the Lagrange multiplier method can be applied based on dot products in the form of $\boldsymbol{x}_{SVMi}^{T}\boldsymbol{x}_{SVMj}$. For a non-linear classification problem, a mathematical mapping $\phi(\cdot)$ can be used to perform a kernel trick

$$K_{SVM}\left(\boldsymbol{x}_{SVMi}, \boldsymbol{x}_{SVMj}\right) = \phi(\boldsymbol{x}_{SVMi})^{T}\phi\left(\boldsymbol{x}_{SVMj}\right) \tag{12}$$

to transform $\boldsymbol{x}_{SVM}$ onto a new vector space. In this study, we applied the commonly used radial basis function (RBF) as the kernel

$$K_{SVM-RBF}\left(\boldsymbol{x}_{SVMi}, \boldsymbol{x}_{SVMj}\right) = \exp\left(-\gamma_{SVM}\|\boldsymbol{x}_{SVMi} - \boldsymbol{x}_{SVMj}\|^2\right), \tag{13}$$

where

$$\gamma_{SVM} = \frac{n}{\sum_{i=1}^{n}\sum_{j=1}^{m}\left(x_{SVMij} - \bar{\boldsymbol{x}}_{SVM}\right)^2} \tag{14}$$

and m was the number of input variables. The SVCs were achieved via the *svc* function of the svm package of scikit-learn library [52].

3.2.4. Artificial neural network classification

In this study, ANNC referred to the method of using an artificial neural network (ANN) for classification. ANN is a model consisting of layers of connected computational units that mimic the functioning of

neurons of an animal brain. The generic architecture of ANN, called a multilayer perceptron (MLP), has an input layer, a set of hidden layers, and an output layer. An ANN is called a deep learning (DL) model when it has at least 2 hidden layers. For the study, we adopted a DL-MLP with 2 hidden layers for ANNC. Each of the input and hidden layers had the same number of nodes as the input variables. The output layer had only 1 node, corresponding to 1 output variable. At a hidden or output layer, each node computed

$$x_{\mathrm{ANN}q,r} = f_{\mathrm{ANN}q,r}\left(\beta_{\mathrm{ANN}q,r,0} + \sum_{s=1}^{N_q-1} \beta_{\mathrm{ANN}q,r,s} x_{\mathrm{ANN}q-1,s}\right), \tag{15}$$

where $x_{\mathrm{ANN}q,r}$ was the output of the rth node of the qth layer, $f_{\mathrm{ANN}q,r}(\cdot)$ was an activation function, $\beta_{\mathrm{ANN}q,r,0}$ and $\beta_{\mathrm{ANN}q,r,s}$ were model parameters, and $N_{q-1} = m$ was the number of nodes of the layer before the qth layer. For ANNC, the activation functions for the hidden and output layers were respectively the rectified linear unit (ReLU) function [53,54]

$$f_{\mathrm{ANN-ReLU}}(x) = \max(0, x) \tag{16}$$

and the logistic sigmoid function

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

The ANNC models were trained with the adaptive moment estimation (Adam) algorithm and the binary cross entropy (BCE) loss function

$$BCE = -\frac{1}{n}\sum_{i=1}^{n}\left[y'_{1i}\ln\left(\hat{y}'_{1i}\right) + \left(1-y'_{1i}\right)\ln\left(1-\hat{y}'_{1i}\right)\right], \tag{18}$$

where y'_{1i} was the simulated observation of binary output and $\hat{y}'_{1i}$ was the binary estimate of the output given by the model with 0.5 as the threshold for classification. The TensorFlow [55] library was used for ANNC modeling.

3.3. Regression models

The regression models in this study were calibrated to estimate the expected loss rate of a data point, given observation of positive loss. The prediction of each regression model was logistically transformed with Eq. (4) to derive the estimate of expected loss. As a demonstration, 4 regression modeling methods were used in the study. They included MLR, RF regression (RFR), support vector regression (SVR), and ANN regression (ANNR).

3.3.1. Multiple linear regression

MLR is the most basic ML, as well as statistical, regression approach that establishes the mathematical relationship between multiple input variables and 1 output variable. As shown in Eq. (5), the MLR method predicts the output as an affine function of the values of input variables. In the study, we applied the *LinearRegression* function of the linear_model package of the scikit-learn library to conduct MLRs [52]. The second part of the true model was also designed as an MLR model for both scenarios (Tables 1 and 2).

3.3.2. Random forest regression

RFR is an ML regression method that uses RF modeling to predict an output numerical value based on input variables. In this study, the *RandomForestRegressor* function of the ensemble package of the scikit-learn library was used for RFR [52]. The loss rate given positive loss of a data point was predicted by the RFR model as the mean of the predictions of the randomly generated RF trees. For RFR, the mean squared error (MSE)

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(\hat{y}_{\mathrm{R}i} - y_{\mathrm{R}i}\right)^2 \tag{19}$$

was used as the optimal split criterion for growing the trees, where $\hat{y}_{\mathrm{R}i}$ was the point estimate of the logit of y_{2i}.

3.3.3. Support vector regression

SVR is an SVM-based method for solving regression problems. Unlike SVC, the SVR hyperplane is in a vector space of data points involving both input and output variables. The hyperplane is located at the center of a margin that contains most of the data points. The support vectors in this case refer to the data points on or near the 2 hyperplanes at the boundaries of the margin. Given data points

$$\{(\boldsymbol{x}_{\mathrm{SVM}1}, y_{\mathrm{SVR}1}), (\boldsymbol{x}_{\mathrm{SVM}2}, y_{\mathrm{SVR}2}), \cdots, (\boldsymbol{x}_{\mathrm{SVM}n}, y_{\mathrm{SVR}n})\} \subset \mathscr{X} \times \mathbb{R}, \tag{20}$$

the SVR hyperplane is

$$y_{\mathrm{SVR}} = \boldsymbol{w}_{\mathrm{SVR}}^{\mathrm{T}}\boldsymbol{x}_{\mathrm{SVM}} + b_{\mathrm{SVR}} \tag{21}$$

where $\boldsymbol{w}_{\mathrm{SVR}} \in \mathscr{X}$ and $b_{\mathrm{SVR}} \in \mathbb{R}$ are model parameters. To derive y_{SVR}, we can minimize

$$L_{\mathrm{SVR}} = \frac{1}{2}\|\boldsymbol{w}_{\mathrm{SVR}}\|^2 + C_{\mathrm{SVR}}\sum_{i=1}^{n}(\xi_{\mathrm{SVR}i} + \zeta_{\mathrm{SVR}i}), \tag{22}$$

subject to

$$\begin{cases} y_{\mathrm{SVR}i} - \boldsymbol{w}_{\mathrm{SVR}}^{\mathrm{T}}\boldsymbol{x}_{\mathrm{SVM}i} - b_{\mathrm{SVR}} \le \varepsilon_{\mathrm{SVR}} + \xi_{\mathrm{SVR}i} \\ \boldsymbol{w}_{\mathrm{SVR}}^{\mathrm{T}}\boldsymbol{x}_{\mathrm{SVM}i} + b_{\mathrm{SVR}} - y_{\mathrm{SVR}i} \le \varepsilon_{\mathrm{SVR}} + \zeta_{\mathrm{SVR}i} \\ \xi_{\mathrm{SVR}i}, \zeta_{\mathrm{SVR}i} \ge 0 \end{cases}, \tag{23}$$

where $\varepsilon_{\mathrm{SVR}}$ is the precision parameter of the margin, C_{SVR} is the regularization hyperparameter of SVR, and $\xi_{\mathrm{SVR}i}$ and $\zeta_{\mathrm{SVR}i}$ correspond to 2 slack variables that allow model calibration with data points being outside the margin. Like SVC, SVR can be conducted with the assistance of a kernel trick. In the study, we also used the RBF kernel for SVR, as described by Eqs. (12)–(14). The *svr* function of the svm package of scikit-learn library was used to perform SVRs [52].

3.3.4. Artificial neural network regression

ANNR in this study refers to the method of ANN modeling for solving a regression problem. We adopted the same architecture of ANNC for ANNR except for the activation function of the output layer and the loss function for model training. The activation function of the output layer for ANNR was the identity function

$$f_{\mathrm{ANN-I}}(x) = x \tag{24}$$

The loss function for model training was the MSE as shown in Eq. (19). To perform ANNR modeling, we used the TensorFlow [55] library.

3.4. Variable selection

For Scenario nV, in this study, a variable selection process was needed to select pertinent VIs from candidate VIs as input variables of ML models, as the true model simulated the loss rates based on the input of 1 IM and 2 VIs (V0 and V1) only. The appropriate selection of pertinent VIs is a key part of modeling vulnerability. In this study, however, as the VIs were randomly generated, they did not explicitly correspond to any real-world VIs. To select the true VIs among the simulated candidates, a number of variable selection methods may be attempted.

Variable selection methods are commonly categorized into 3 groups, i.e., the filter, embedded, and wrapper methods [56]. Firstly, a filter method selects variables based on the relationships between the candidate variables without consideration of the output variable. Because the variable selection by a filter method is irrelevant to the mathematical mapping between input and output variables, such a method is inappropriate for our purposes. Next, an embedded method refers to a model calibration process that simultaneously performs variable selection. An

example of embedded method for variable selection is Lasso (see, e.g., [11,57,58]). Because the embedded method can only be used for a limited number of specific ML approaches, it cannot be applied universally for ML methods such as the ones adopted in this study. Lastly, a wrapper method selects the optimal combination of input variables by comparing the predictive performances of models calibrated with different combinations of input variables. For example, wrapper methods can be applied with an algorithm of brute force [42,48] or stepwise deletion [43]. Due to calibrations of the examined models with different combinations of input variables, the computational time for a wrapper method can be exceptionally large, especially when it is applied to a computationally expensive ML method, such as RF, RBF-SVM, and DL-ANN.

To overcome the limitations of the existing variable selection methods, we proposed, in this study, a permutation importance algorithm (PIA) for identification of pertinent variables of VIs for the suggested ML-EPMs in Scenario nV. PIA is an algorithm that derives a score of relative importance for each input variable in a model based on the change in model performance on testing data when the testing values of the variable is randomly shuffled [58,59]. The PIA for the study had 12 steps. First, the training dataset for an ML-EPM was randomly separated into a training-training dataset with 90% of the TDPs and a training-testing dataset with the remaining 10% of the TDPs. Second, the ML-EPM was calibrated on the training-training data. Third, a loss was computed on the training-testing data as a reference score (RS). For classification and regression, the RSs were respectively the BCE as in Eq. (18) and the root MSE (RMSE)

$$RS_{\mathrm{RMSE}} = \sqrt{\frac{1}{n}\sum\nolimits_{i=1}^{n}(\widehat{y}_{\mathrm{R}i} - y_{\mathrm{R}i})^2}. \tag{25}$$

Fourth, the first input variable was selected if it had not been examined. Otherwise, the next input variable was selected for examination. Fifth, the values of the selected variable were shuffled to generate a corrupted version of the training-testing data. Sixth, a variable shuffle score (VSS) was computed as equal to the loss measure of the model on the corrupted training-testing data. Seventh, an importance score (IS) was derived with

$$IS = VSS - RS. \tag{26}$$

Eighth, steps 5–7 were repeated 32 times to produce 32 ISs. Ninth, steps 4–8 were repeated to cover all the input variables. Tenth, one-tailed one-sample *t*-tests were conducted on the ISs of input variables with the null hypotheses that the ISs were supposed to be no greater than 0. Eleventh, an input variable was selected when the null hypothesis on it was rejected at the significance level of 10^{-17}. This significance value was determined to be optimal for variable selection in this study based on trials and errors. Twelfth, the ML-EPM was recalibrated with the selected input variables on the entire training dataset.

3.5. Model validation

To compare the predictive performances of the calibrated ML-EPMs, we applied the models to the testing dataset for validation. For classification models, we used the validation metrics of false positive rate (FPR), precision (Pr), recall (Re), and F1 score (F1):

$$FPR = \frac{FP}{FP + TN}, \tag{27}$$

$$Pr = \frac{TP}{TP + FP}, \tag{28}$$

$$Re = \frac{TP}{TP + FN}, \tag{29}$$

$$F1 = \frac{2PrRe}{Pr + Re}, \tag{30}$$

where *TP*, *FP*, *TN*, and *FN* were respectively the numbers of true positives, false positives, true negatives, and false negatives. For validating regression models, we applied the metrics of mean error (ME), mean absolute error (MAE), RMSE, and mean absolute percentage error (MAPE):

$$ME = \frac{1}{n}\sum_{i=1}^{n}(\widehat{y}_{2i} - y_{2i}), \tag{31}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|\widehat{y}_{2i} - y_{2i}|, \tag{32}$$

$$RMSE = \sqrt{\frac{1}{n}\sum\nolimits_{i=1}^{n}(\widehat{y}_{2i} - y_{2i})^2}, \tag{33}$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{\widehat{y}_{2i} - y_{2i}}{y_{2i}}\right|, \tag{34}$$

where $\widehat{y}_{2i}$ was the point estimate of loss rate and $|\cdot|$ was the absolute value operator.

4. Results and discussions

With the implementation of the proposed MC experimental methodology, we may produce slightly different results with different runs of the codes for the study. Albeit slightly different, these results consistently pointed toward the same conclusions. In this paper, we present only one set of these results as a demonstration. The results included the validation of individual ML-EPMs, generation of vulnerability curves, and comparisons of model sensitivity to the variation of number of TDPs.

4.1. Validation results

The validation of the examined ML-EPMs involved both Scenarios 1V and nV. For Scenario 1V, the numbers of TDPs for the classification and regression models were 10,000 and 2809, corresponding to all TDPs and the TDPs containing positive loss rates, respectively. With these TDPs, all 8 individual ML-EPMs performed almost as well as their corresponding true models, as shown in Fig. 4. Regarding classification, the receiver operating characteristic (ROC) curves of the calibrated LR, RFC, SVC, and ANNC models were almost identical to the one produced by the true LR model (Fig. 4(a)–(d)). Some of the classification EPMs trained with all 10,000 TDPs performed even better than the true model in terms of some of the validation metrics, as shown in Table 3. For example, the ANNC model produced a better FPR (0.0671) than the true model (0.0711). The LR (0.6334) and ANNC (0.6401) models resulted in higher precisions than the true model (0.6333). Meanwhile, the LR (0.4343), RFC (0.4348), and SVC (0.4716), especially SVC, models yielded higher F1 scores than the true model (0.4320). These better performances in F1 scores were achieved because the LR (0.3304), RFC (0.3370), and SVC (0.3917) models produced much higher recalls than the true model (0.3278).

Noteworthily, as in Table 3, all classification models, including the true model, produced a much higher precision than recall. As a result, the F1 scores derived with such precisions and recalls would tend to be smaller than the one computed when a classification EPM was trained to produce a precision and recall that were almost equal to each other. This is the reason why the SVC model, among all models trained with all 10,000 TDPs plus the true model, produced the highest F1 score, as its precision and recall were the closest to each other compared to the ones produced by the other models. The results of the unbalanced values of precisions and recalls leading to relatively low F1 scores in this study was largely because of the unbalanced distribution of TDPs with 0 losses (7191 TDPs) and positive losses (2809 TDPs). To overcome this issue of unbalanced numbers of TDPs with different labels, we may apply a

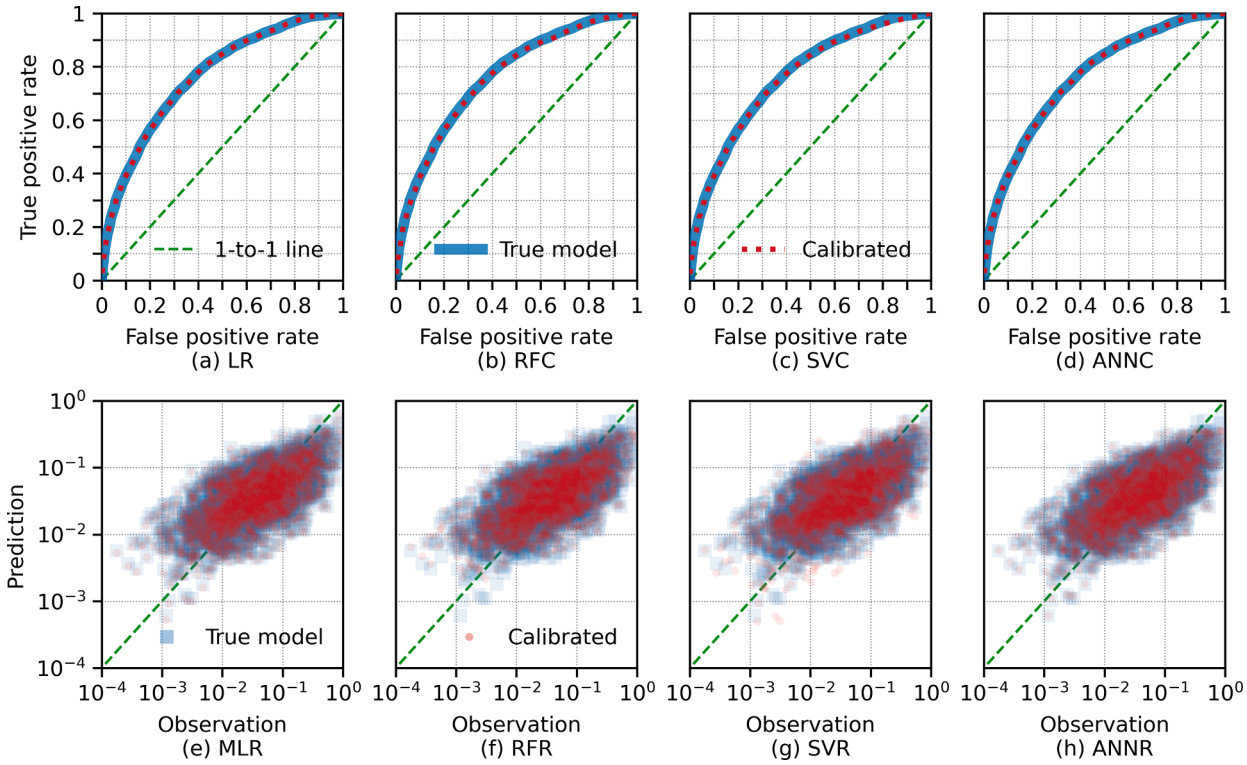


Fig. 4. Results of validation of ML-EPMs for Scenario 1V, including comparisons of ROC curves of the (a) LR, (b) RFC, (c) SVC, and (d) ANNC models and distributions of data points predicted by the (e) MLR, (f) RFR, (g) SVR, and (h) ANNR models.

Table 3
Statistics of model validation and sensitivity analysis for classification on testing data for Scenario 1V.

Model	TDPs	FPR	Precision	Recall	F1
True	N/A	0.0711	0.6333	0.3278	0.4320
LR	10,000	0.0716	*0.6334*	*0.3304*	*0.4343*
	1000	0.0744	0.6287	*0.3363*	*0.4382*
	500	0.0869	0.6070	*0.3583*	*0.4506*
RFC	10,000	0.0799	0.6124	*0.3370*	*0.4348*
	1000	0.0745	0.6129	0.3150	0.4161
	500	*0.0613*	*0.6359*	0.2860	0.3945
SVC	10,000	0.1009	0.5924	***0.3917***	***0.4716***
	1000	*0.0241*	*0.7231*	0.1678	0.2723
	500	0.1284	0.5660	*0.4471*	*0.4996*
ANNC	10,000	***0.0671***	***0.6401***	0.3186	0.4255
	1000	0.0842	0.6149	*0.3594*	*0.4537*
	500	0.0818	0.6149	*0.3488*	*0.4451*

Note: Italic indicates better performance than the true model; **bold** indicates best performance regarding a metric among models trained with all 10,000 TDPs plus the true model.

Table 4
Statistics of model validation and sensitivity analysis for regression on testing data for Scenario 1V.

Model	TDPs	ME	MAE	RMSE	MAPE
True	N/A	**–0.0260**	**0.0524**	**0.0979**	1.4556
MLR	2809	–0.0275	0.0525	0.0984	*1.3950*
	500	–0.0310	0.0529	0.0996	*1.2506*
	250	–0.0288	0.0530	0.0992	*1.3506*
RFR	2809	–0.0291	0.0531	0.1003	*1.4295*
	500	–0.0389	0.0561	0.1098	*1.3085*
	250	–0.0388	0.0577	0.1132	1.4950
SVR	2809	–0.0278	0.0531	0.1001	***1.3858***
	500	–0.0315	0.0546	0.1048	*1.2889*
	250	–0.0319	0.0572	0.1102	*1.4042*
ANNR	2809	–0.0262	0.0525	0.0981	*1.4191*
	500	–0.0319	0.0533	0.1014	*1.2709*
	250	–0.0286	0.0534	0.0999	*1.3590*

Note: Italic indicates better performance than the true model; **bold** indicates best performance regarding a metric among models trained with 2809 TDPs plus the true model.

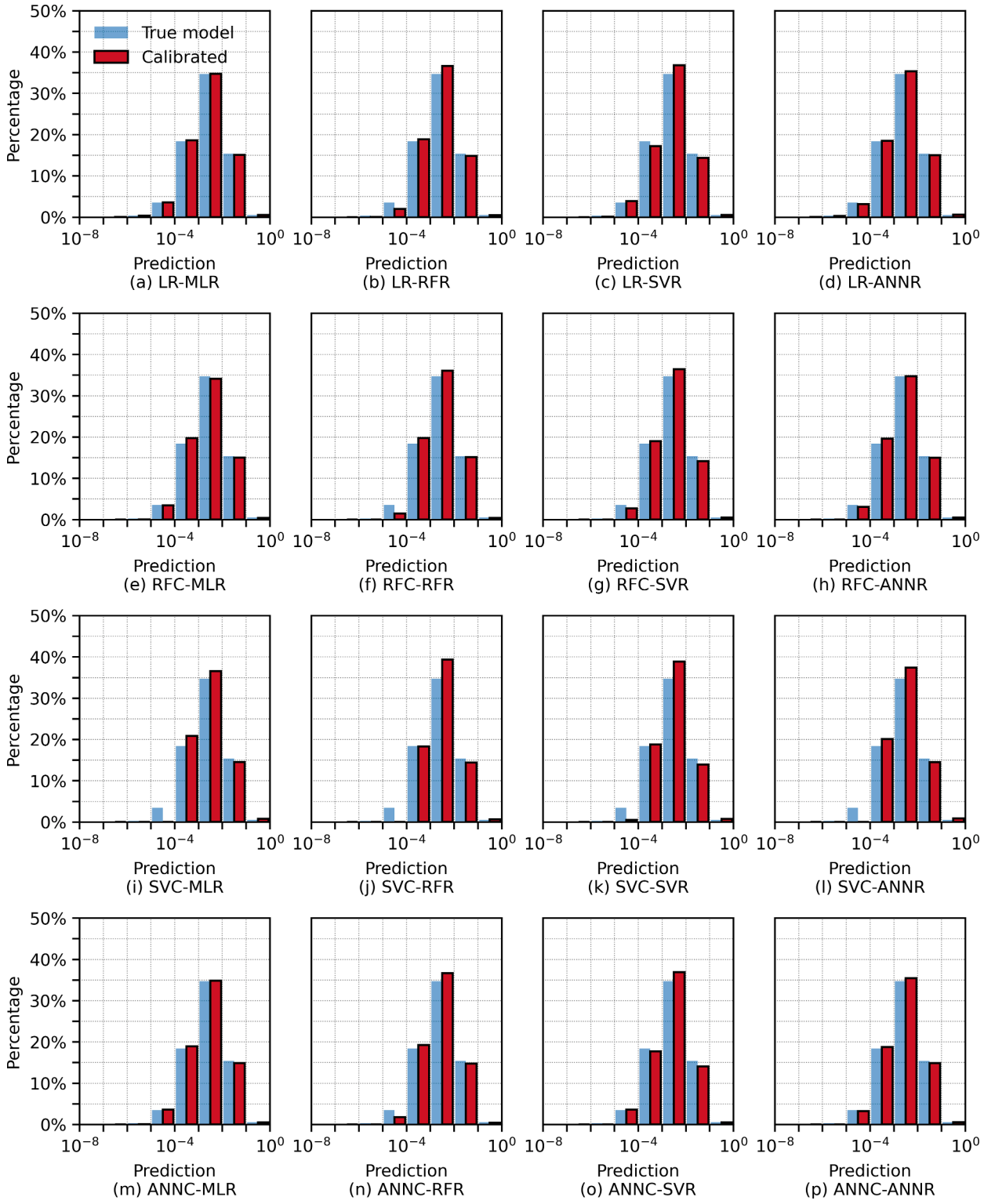


Fig. 5. Distributions of predicted loss rates on testing data points with 0 simulated losses produced by the joint calibrated models of (a) LR-MLR, (b) LR-RFR, (c) LR-SVR, (d) LR-ANNR, (e) RFC-MLR, (f) RFC-RFR, (g) RFC-SVR, (h) RFC-ANNR, (i) SVC-MLR, (j) SVC-RFR, (k) SVC-SVR, (l) SVC-ANNR, (m) ANNC-MLR, (n) ANNC-RFR, (o) ANNC-SVR, and (p) ANNC-ANNR in Scenario 1V.

subsampling technique to balance the numbers of TDPs, as suggested by previous studies (see, e.g., Wang and Sebastian, [60]). Interestingly, however, unlike the previous efforts that did not use a true model for comparison of predictive performances, we noticed in this study that, once introduced, the true model for classification also yielded unbalanced values of precision and recall. This result indicated that it may not be necessary to balance the numbers of TDPs with different labels to train the EPMs for comparison of predictive performances of the trained models versus the performance of the true model.

Regarding the calibrated regression EPMs, in particular the calibrated MLR, SVR, and ANNR models, their predictions of loss rates on the testing data showed little difference from the ones by the true MLR

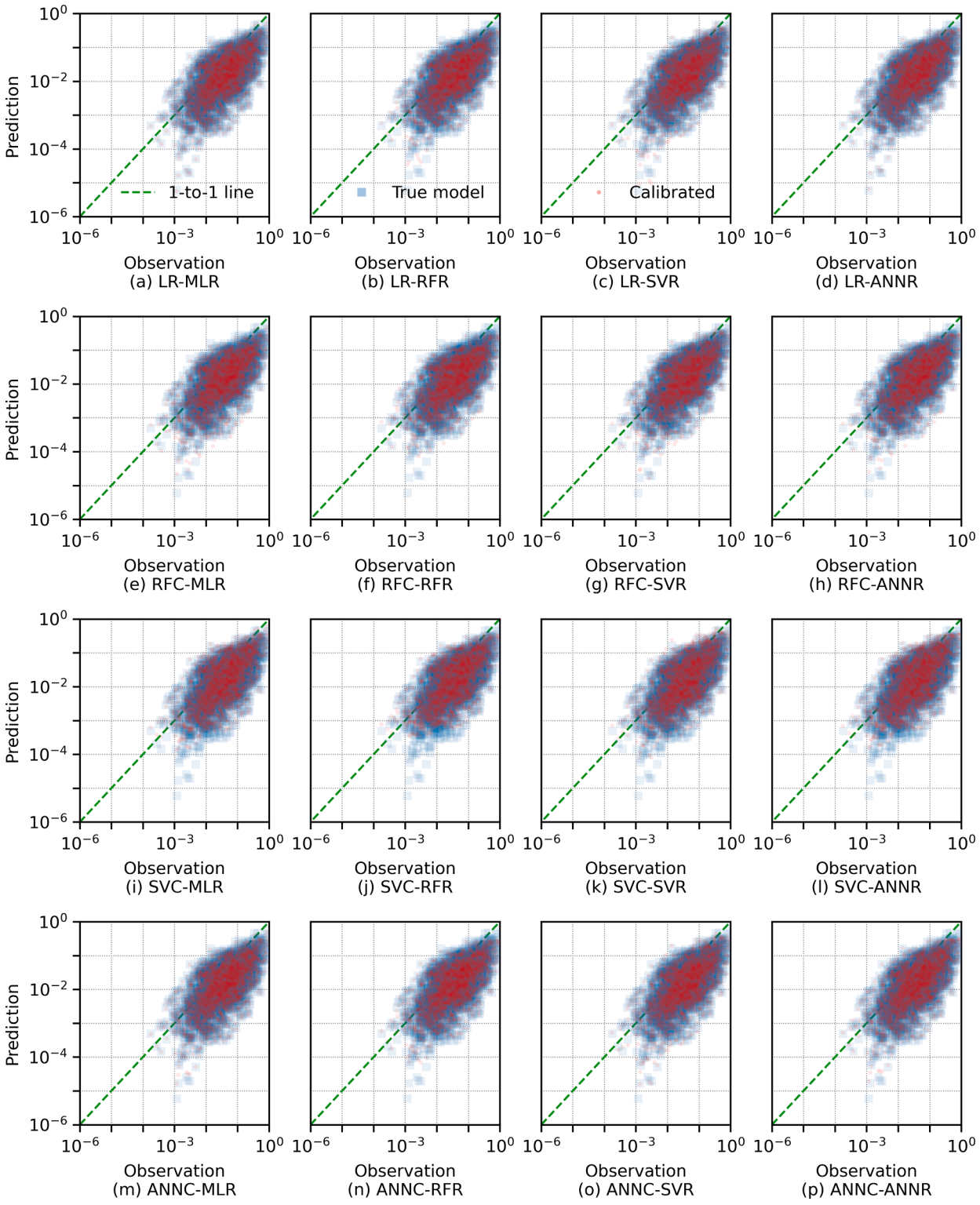


Fig. 6. Simulated observations of positive loss rates versus the predictions produced by the joint calibrated models of (a) LR-MLR, (b) LR-RFR, (c) LR-SVR, (d) LR-ANNR, (e) RFC-MLR, (f) RFC-RFR, (g) RFC-SVR, (h) RFC-ANNR, (i) SVC-MLR, (j) SVC-RFR, (k) SVC-SVR, (l) SVC-ANNR, (m) ANNC-MLR, (n) ANNC-RFR, (o) ANNC-SVR, and (p) ANNC-ANNR in Scenario 1V.

model (Fig. 4(e)–(h)). Despite the similar model performances, however, the calibrated regression EPMs did not outperform the true model in terms of the validation metrics except for MAPE, as listed in Table 4. Among the 4 regression EPMs trained with all 2809 TDPs, the ANNR model resulted in the smallest ME (–0.0262) and RMSE (0.0981). In terms of the MAE, the MLR and ANNR models produced almost identical performances (0.0525) that were slightly better than the ones of the RFR and SVR models (0.0531). Regarding MAPE, the SVR model yielded the best result (1.3858), while all trained regression EPMs produced MAPEs lower than the one of the true model (1.4556).

For Scenario 1V, the joint calibrated EPMs made similar predictions of loss rates on the testing data as the joint true model (Figs. 5 and 6).

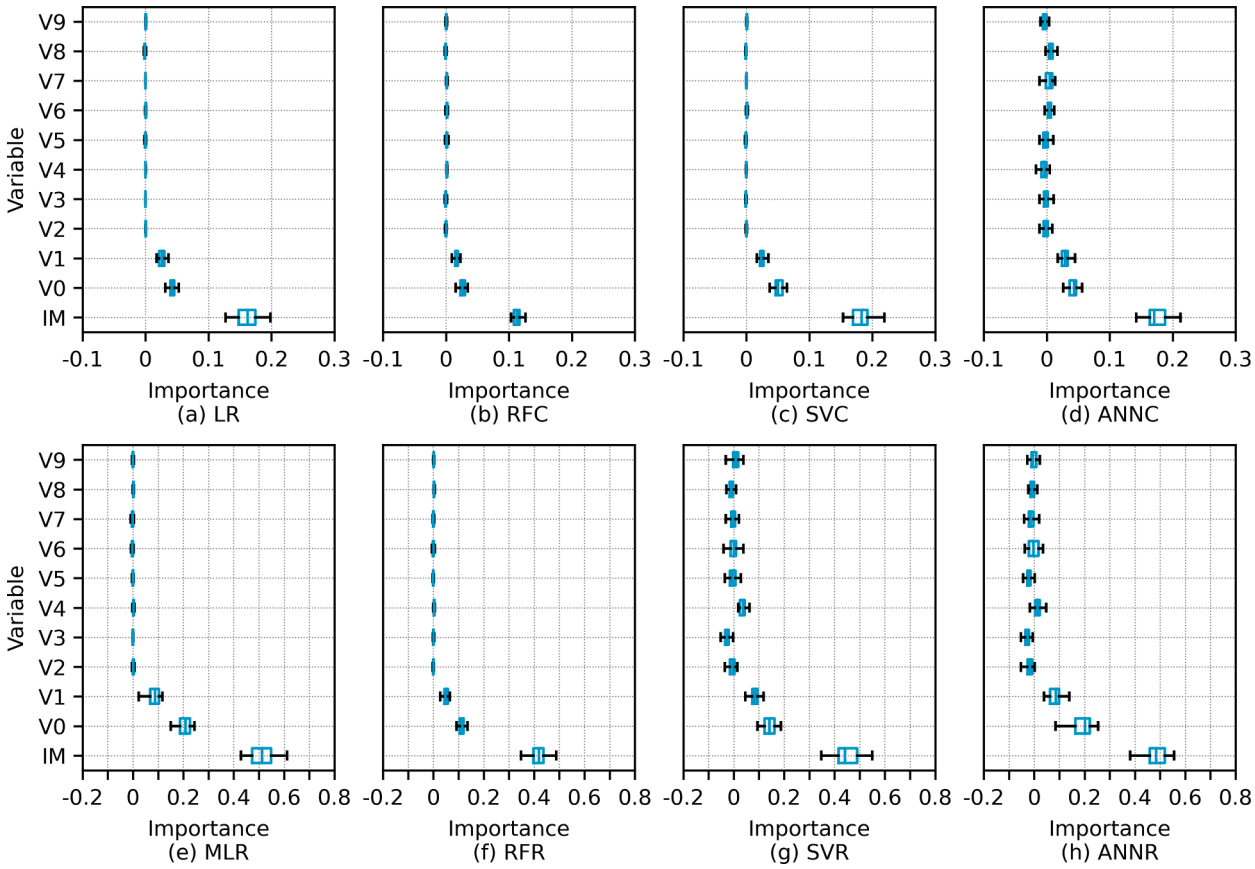


Fig. 7. Importance measures of the input variables of the calibrated (a) LR, (b) RFC, (c) SVC, (d) ANNC, (e) MLR, (f) RFR, (g) SVR, and (h) ANNR models in Scenario nV.

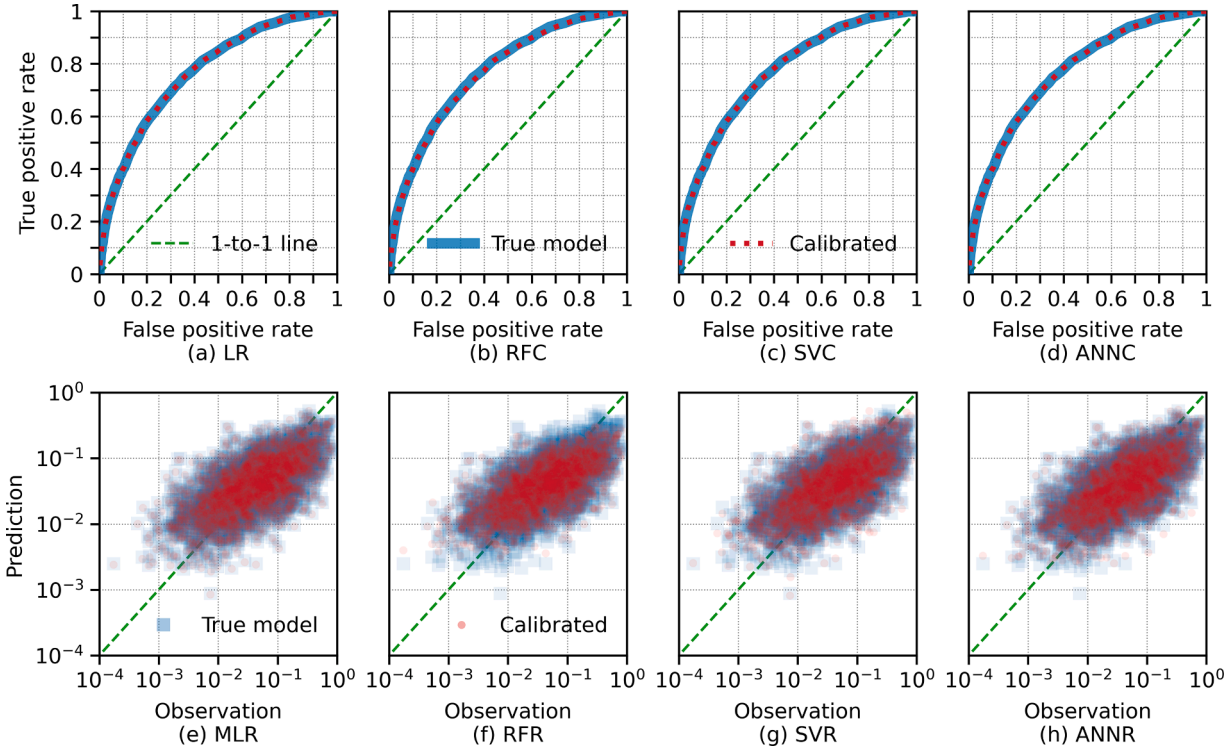


Fig. 8. Results of validation of ML-EPMs for Scenario nV, including comparisons of ROC curves of the (a) LR, (b) RFC, (c) SVC, and (d) ANNC models and distributions of data points predicted by the (e) MLR, (f) RFR, (g) SVR, and (h) ANNR models.

Although many simulated data points contained 0 losses, the model predictions of loss rates were always positive, as formulated in Eq. (6), even for the true model. This resulted in a slight overestimation of loss rates over the data points with 0 simulated losses. Despite such an overestimation of loss rates, the distributions of the overestimated loss rates on the 0 data points by the joint calibrated models, especially the calibrated LR-MLR, LR-ANNR, RFC-MLR, RFC-ANNR, ANNC-MLR, and ANNC-ANNR models, were highly similar to the one produced by the joint true model, as shown in Fig. 5.

As also shown in Eq. (6), for a testing data point with a positive simulated loss, the prediction of loss rate by a regression model was multiplied by an expected probability with a range of $(0, 1)$ to provide the joint-model prediction of loss rate. Thus, the joint models, including the joint true model, always tended to provide predictions that underestimated the loss rates of data points with positive simulated losses, as shown in Fig. 6. This underestimation on the positive data points compensated for the effect of the overestimation on the 0 data points so that the joint models tended to be minimally biased in general (see the ME column in Table 4). With Figs. 5 and 6, we noticed that the joint calibrated models predicted the loss rates of data points in almost the same way as the joint true model.

For Scenario nV, in addition to the predictive performances of the classification, regression, and joint models, the PIA for variable selection was also tested for the ML-EPMs. In this scenario, 10,000 and 2836 TDPs were used to train the classification and regression EPMs, respectively. Meanwhile, both the true models for classification and regression only included the variables of IM, V0, and V1 as their input variables. Like in most of the attempted runs of codes of the study, in this presented run, the PIA successfully selected the same 3 variables as the only input variables of all 8 ML-EPMs. As shown in Fig. 7, however, there seemed to be more uncertainties involved regarding the non-pertinent input variables in the variable selection processes with the PIA for the ANNC (Fig. 7(d)), SVR (Fig. 7(g)), and ANNR (Fig. 7(h)) models than with the other models. In addition, the RFC (Fig. 7(b)) and RFR (Fig. 7(f)) models tended to result in smaller values of importance scores than the other models. Because the VIs used in this study were stochastically generated to mimic, but without explicit association with, real-world indicators, future study should further examine if the similar comparative results in PIA applications to different ML-EPMs for variable selection also hold true for real-world data.

Like in Scenario 1V, the predictive performances of the examined ML-EPMs in Scenario nV were highly similar to the one of the true model, as indicated by Fig. 8 and Tables 5 and 6. As shown in Fig. 8(f), however, the RFR model seemed to overestimate the smaller observations and underestimate the larger observations more significantly than the other regression EPMs and the true model. This discernible poor predictive performance of the RFR model was also indicated by the worst ME and RMSE of the RFR model as listed in Table 6. As indicated by Table 5, when the selection of input variables was involved, the calibrated classification EPMs tended to produce higher FPRs than the true model. Similar to Scenario 1V, in Scenario nV, when the precision and recall derived with an EPM were closer to each other than the ones yielded by the true model, the F1 score of the EPM tended to be larger than the one of the true model, even though the true model may result in a much better precision score (see Table 5). Regarding the regression predictive performances, the true model seemed to yield better ME (–0.0301), MAE (0.0580), and RMSE (0.1057) than the calibrated regression EPMs, except for the SVR model that resulted in a better ME (–0.0296) than the true model. In terms of MAPE, apart from SVR, all the other regression EPMs produced a better score than the true model (1.3646). In particular, the ANNR model resulted in the best MAPE (1.3354).

Table 5
Statistics of model validation for classification on testing data for Scenario nV.

Model	FPR	Precision	Recall	F1
True	**0.0779**	**0.6303**	0.3535	0.4529
LR	0.0845	0.6219	*0.3700*	*0.4639*
RFC	0.0801	0.6276	*0.3593*	*0.4570*
SVC	0.1206	0.5798	***0.4432***	***0.5024***
ANNC	0.0840	0.6221	*0.3685*	*0.4628*

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

Table 6
Statistics of model validation for regression on testing data for Scenario nV.

Model	ME	MAE	RMSE	MAPE
True	–0.0301	**0.0580**	**0.1057**	1.3646
MLR	–0.0306	0.0580	0.1059	*1.3437*
RFR	–0.0356	0.0588	0.1094	*1.3619*
SVR	***–0.0296***	0.0592	0.1072	1.3768
ANNR	–0.0305	0.0582	0.1060	***1.3354***

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

In Scenario nV, the joint calibrated models also provided predictions of loss rates in a highly similar manner as the joint true model, as shown in Figs. 9 and 10. Among all the joint calibrated models, the ones with the RFR model, including the LR-RFR (Figs. 9(b) and 10(b)), RFC-RFR (Figs. 9(f) and 10(f)), SVC-RFR (Figs. 9(j) and 10(j)), and ANNC-RFR (Figs. 9(n) and 10(n)), seemed to result in the largest degrees of overestimation and underestimation of the observed loss rates. This result was consistent with the relatively poor predictive performance of the RFR model discussed previously. Despite the slight individual differences in predictive performances of the examined ML-EPMs, in general, the results of model validation of this study suggested that the empirical predictive modeling approach with applications of ML methods can be effectively and accurately used to compute societal vulnerability in the form of an expected loss rate given pre-event societal indicators and the IM of a hazardous event.

4.2. Vulnerability curves

With the calibrated ML-EPMs, we could create vulnerability curves for simulated societal systems for both Scenarios 1V and nV, as demonstrated in Figs. 11 and 12, respectively. Each vulnerability curve represented a function of pertinent VI or VIs given a spectrum of IM for a societal system corresponding to a data point. Here, the big circle referred to the simulated loss rate given IM for the associated testing data point of societal system with the true models. The small triangles corresponded to the TDPs. The wide solid curves were the vulnerability curves produced by the true models. The 99th (Fig. 11) and 117th (Fig. 12) simulated societal systems were chosen to provide a good visual presentation such that the simulated loss rates were not located too close to the bottom-left corner. The IM on the horizontal axis referred to an abstract measure that was exponentially transformed from the simulated logit of IM. The IM in Figs. 11 and 12 was visualized in such a manner to mimic the range of a real-world IM. For example, the range of IM here was similar to that of PGA $\times$ 0.01 for earthquake or water depth for flood with the unit of m s^{-2} or dm, respectively. For IM with a small value such as within a range of $[0, 5]$ in Figs. 11 and 12, the vulnerability curves generated with the joint calibrated models looked similar as the one of the true model. This range of IM was also the range of IMs of the majority of the TDPs. When looking at the larger IM values, however, we noticed that, apart from the calibrated LR-MLR model, all other joint calibrated models tended to produce somewhat poor extrapolations of vulnerability curves for at least one of the two considered scenarios. Given that the joint true model was also an LR-MLR model, this result favoring the calibrated LR-MLR model was not surprising.

For both scenarios, the joint calibrated models presented specific patterns in generating vulnerability curves. For example, all the joint

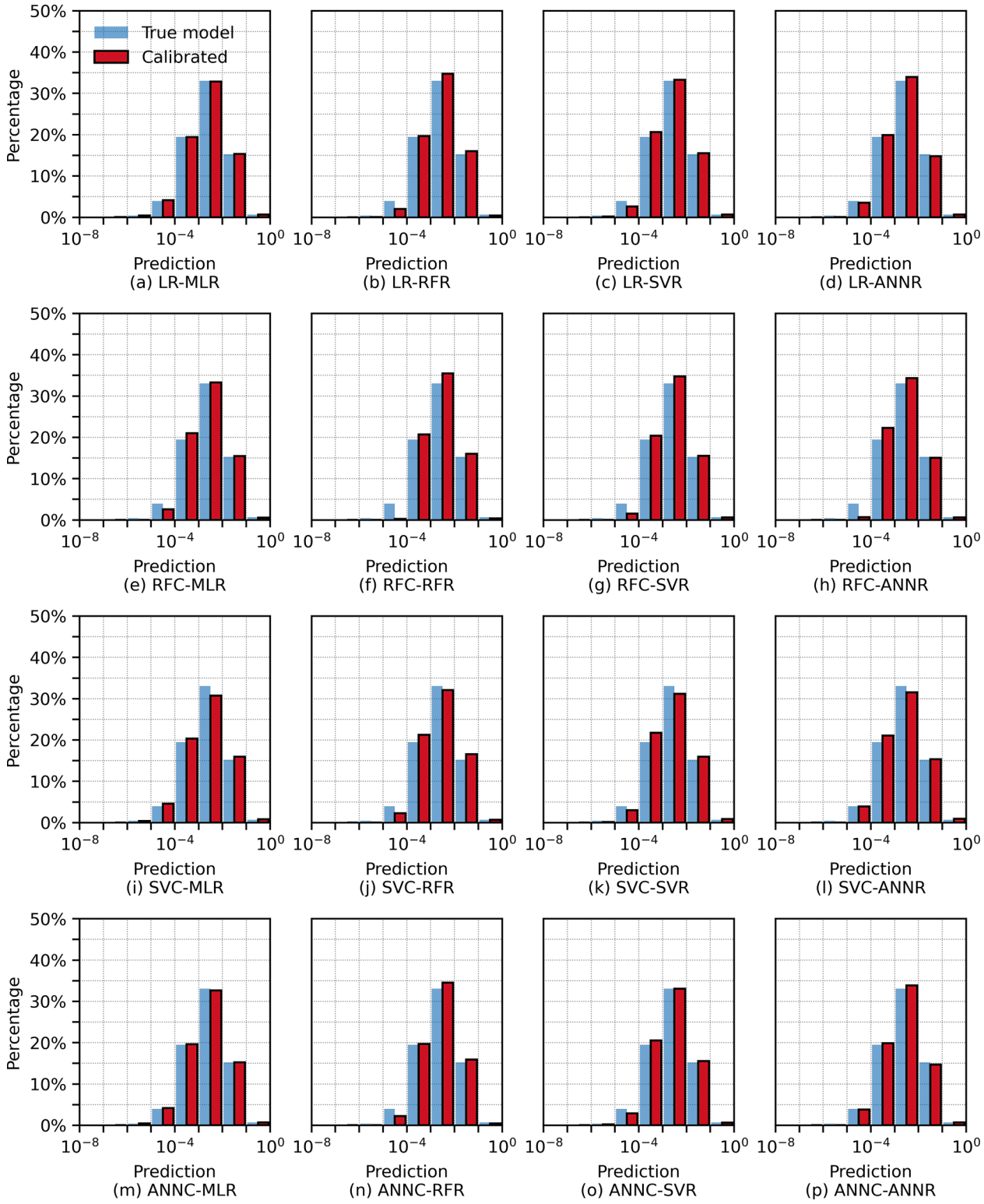


Fig. 9. Distributions of predicted loss rates on testing data points with 0 simulated losses produced by the joint calibrated models of (a) LR-MLR, (b) LR-RFR, (c) LR-SVR, (d) LR-ANNR, (e) RFC-MLR, (f) RFC-RFR, (g) RFC-SVR, (h) RFC-ANNR, (i) SVC-MLR, (j) SVC-RFR, (k) SVC-SVR, (l) SVC-ANNR, (m) ANNC-MLR, (n) ANNC-RFR, (o) ANNC-SVR, and (p) ANNC-ANNR in Scenario nV.

calibrated models using an RFR model for regression tended to produce vulnerability curves that began to flatten significantly after the IM reached beyond 10 (Figs. 11(b), (f), (j), (n), 12(b), (f), (j), and (n)). In addition, these vulnerability curves by RFR-based models showed zigzags for small IM values, indicating overfitting with the RFR modeling. As another example, the extrapolations of vulnerability curves for large IM values by the SVR-based joint models were also not ideal (Figs. 11(c), (g), (k), (o), 12(c), (g), (k), and (o)). Although the vulnerability curves produced by these models tended to follow the vulnerability curves by the true model at small IM values, these curves tended to first overestimate the expected loss rate and then to decline with the increase of IM. The unique shapes of the vulnerability curves derived with the RFR

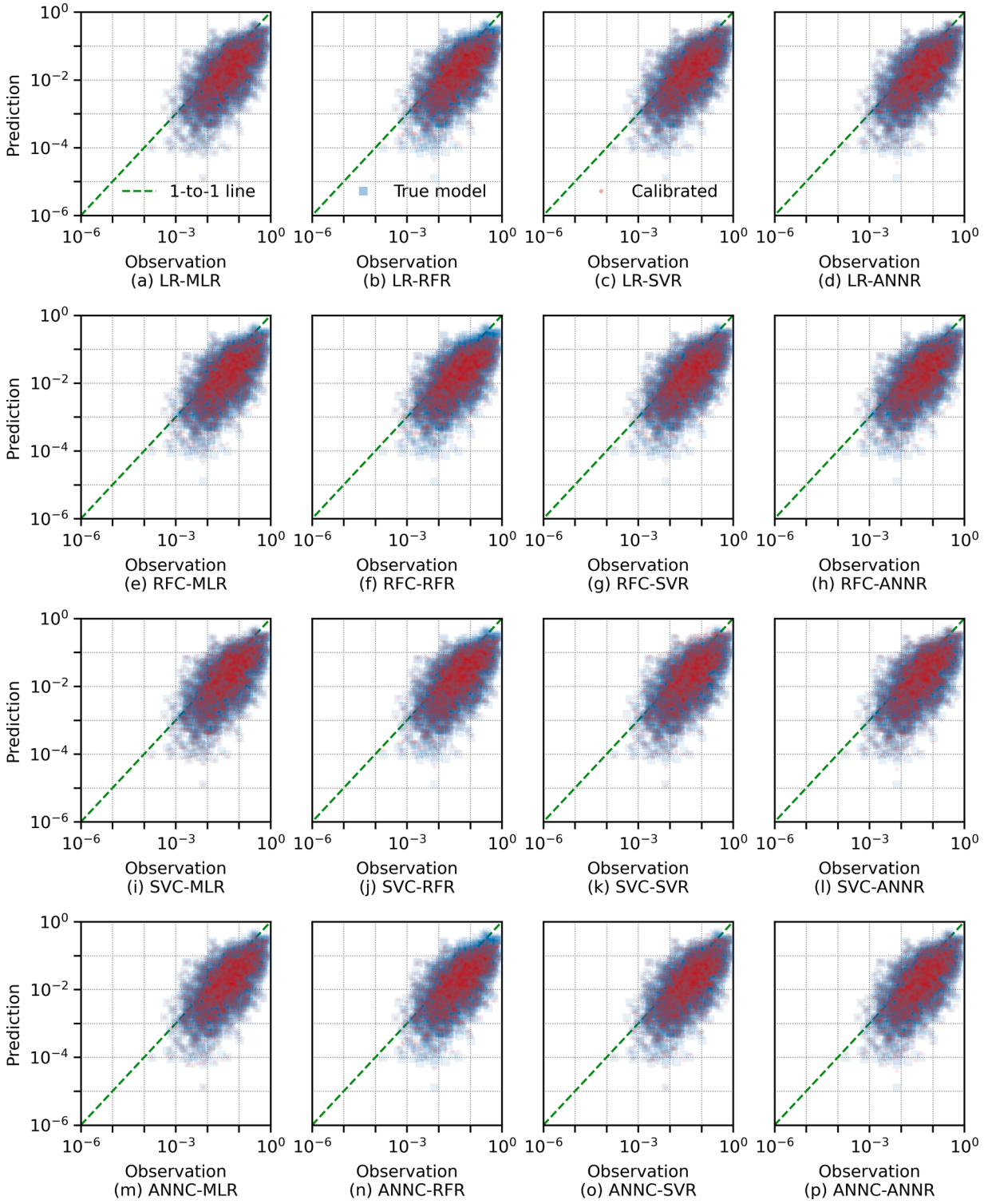


Fig. 10. Simulated observations of positive loss rates versus the predictions produced by the joint calibrated models of (a) LR-MLR, (b) LR-RFR, (c) LR-SVR, (d) LR-ANNR, (e) RFC-MLR, (f) RFC-RFR, (g) RFC-SVR, (h) RFC-ANNR, (i) SVC-MLR, (j) SVC-RFR, (k) SVC-SVR, (l) SVC-ANNR, (m) ANNC-MLR, (n) ANNC-RFR, (o) ANNC-SVR, and (p) ANNC-ANNR in Scenario nV.

and SVR-based joint models were associated with the sophisticated mathematical assumptions of the RFR and SVR models, respectively.

Meanwhile, the ANNR-based joint models produced better vulnerability curves than the ones by the RFR or SVR-based joint models in particular in Scenario nV (Fig. 12(d), (h), (l), and (p)), although these ANNR-based curves also tended to flatten at large IM values in Scenario 1V (Fig. 11(d), (h), (l), and (p)). Despite the DL design, the MLP architectures of the ANNR models in this study were relatively simple and straightforward, given the small number of hidden layers and computational nodes. Such model simplicity may be the reason why the ANNR-based joint models produced better vulnerability curves than the ones by the more sophisticated models.

Regarding the calibrated classification EPMs, they did not show significant differences in affecting the shapes of vulnerability curves,

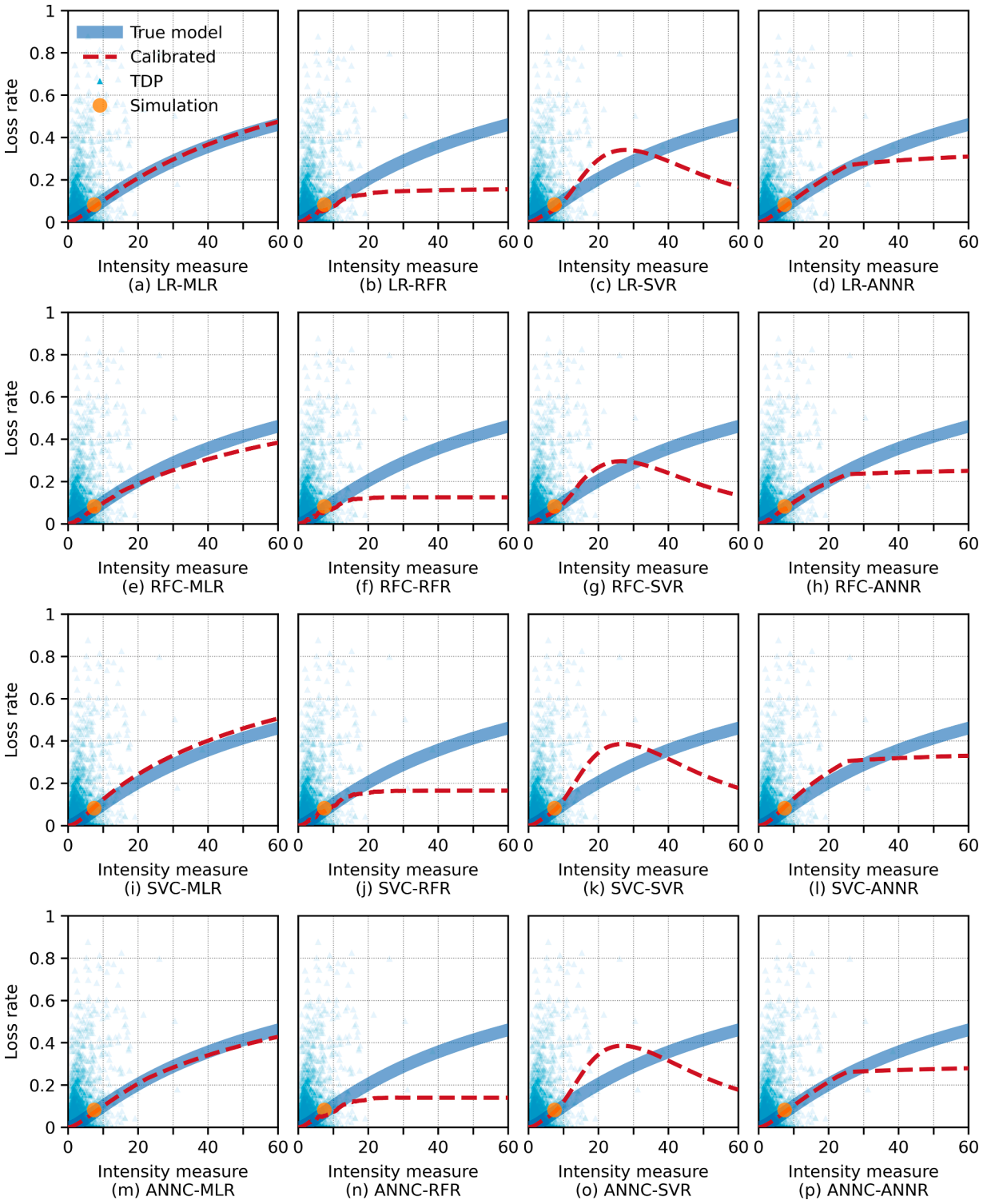


Fig. 11. Vulnerability curve for the 99th simulated testing societal system for Scenario 1V produced by the calibrated models of (a) LR-MLR, (b) LR-RFR, (c) LR-SVR, (d) LR-ANNR, (e) RFC-MLR, (f) RFC-RFR, (g) RFC-SVR, (h) RFC-ANNR, (i) SVC-MLR, (j) SVC-RFR, (k) SVC-SVR, (l) SVC-ANNR, (m) ANNC-MLR, (n) ANNC-RFR, (o) ANNC-SVR, and (p) ANNC-ANNR.

especially in Scenario 1V (Fig. 11(a), (e), (i), and (m)). However, the RFC (Figs. 11(e)–(h) and 12(e)–(h)) and ANNC-based (Figs. 11(m)–(p) and 12(m)–(p)) joint models tended to produce vulnerability curves that slightly underestimated the expected loss rate of the true model, while the SVC-based (Figs. 11(i)–(l) and 12(i)–(l)) joint models tended to generate vulnerability curves with a slight overestimation. These results indicated that the ML methods may be better at solving the classification problems than the regression ones for empirically estimating societal vulnerability. For generating vulnerability curves, a simpler model, especially a simpler regression model, tended to perform better when extrapolation of expectation of loss rate was needed for an IM value beyond its normal range.

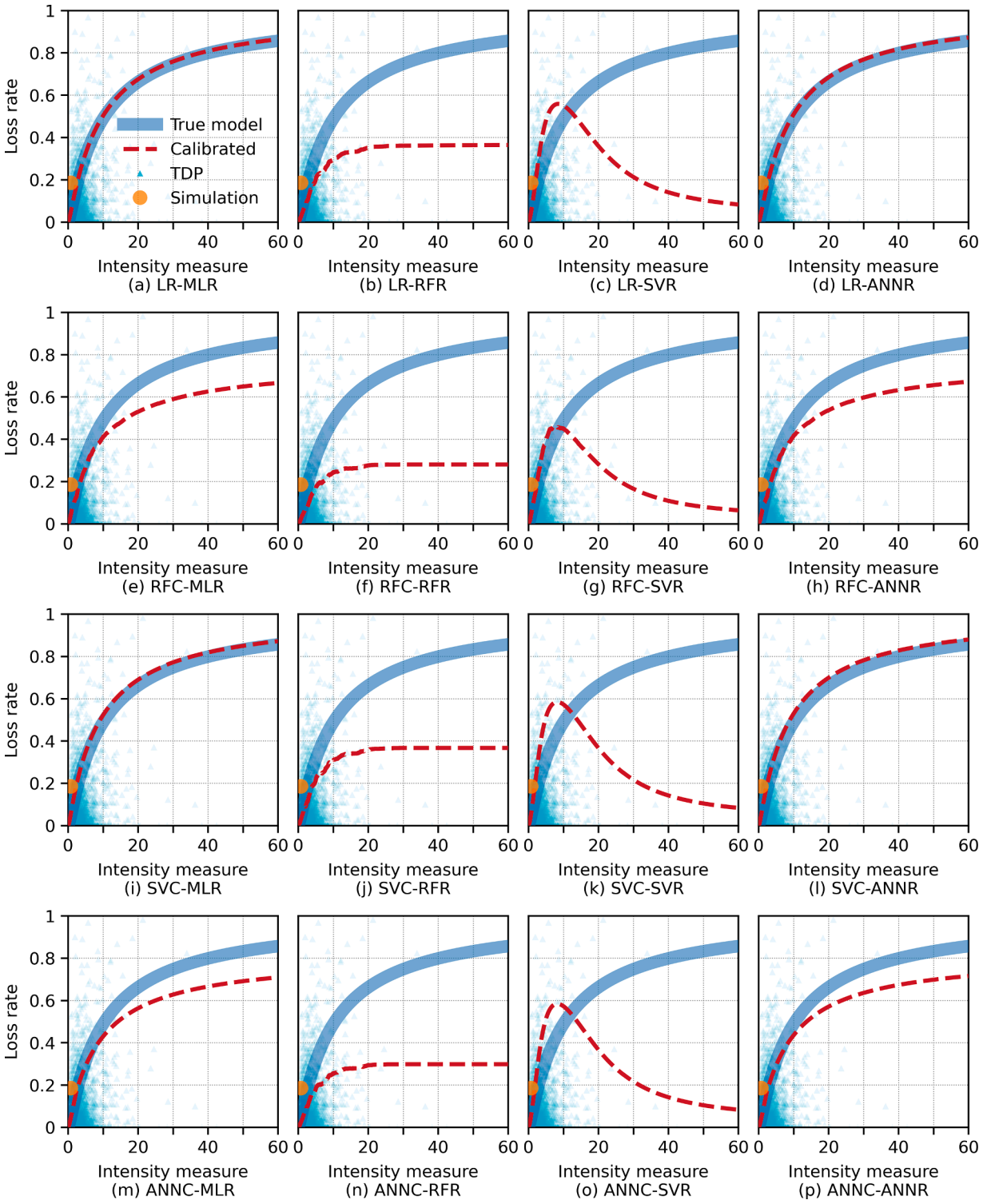


Fig. 12. Vulnerability curve for the 117th simulated testing societal system for Scenario nV produced by the calibrated models of (a) LR-MLR, (b) LR-RFR, (c) LR-SVR, (d) LR-ANNR, (e) RFC-MLR, (f) RFC-RFR, (g) RFC-SVR, (h) RFC-ANNR, (i) SVC-MLR, (j) SVC-RFR, (k) SVC-SVR, (l) SVC-ANNR, (m) ANNC-MLR, (n) ANNC-RFR, (o) ANNC-SVR, and (p) ANNC-ANNR.

4.3. Sensitivity analysis

To examine how the number of TDPs affected the calibrations, variable selections, and predictive performances of the adopted ML-EPMs, we conducted a sensitivity analysis. For sensitivity analysis for classification in Scenario 1V, 1000 and 500 TDPs were randomly selected from the original 10,000 TDPs without replacement to form two datasets. Similarly, for regression in Scenario 1V, 500 and 250 TDPs were randomly selected from the original 2809 TDPs. For classification in Scenario nV, we selected 5000 and 2000 TDPs from the original 10,000 TDPs to form training datasets for sensitivity analysis. For regression in Scenario nV, we selected 2000 and 1000 TDPs from the original 2836 TDPs.

With Scenario 1V, we first examined the effects of number of TDPs on

Table 7
Estimated parameters of calibrated LR and MLR models for Scenario 1V.

Model	TDPs	Model coefficients			σ_{MLR}
		Intercept	IM	VI	
LR	10,000	**–1.1694**	0.9773	**0.4935**	N/A
	1000	–1.2491	**1.0202**	0.5948	N/A
	500	–1.3269	1.1307	0.7041	N/A
MLR	2809	**–4.0566**	**1.0286**	**0.4845**	1.2170
	500	–4.2305	1.1279	0.4373	**1.1858**
	250	–4.1288	1.1065	0.3874	1.1838

Note: **Bold** indicates estimate closest to the value of the true model parameter.

calibration of the LR and MLR models with respect to the parameters of the true models described in Section 3.4 and Table 1. In the presented run, as shown in Table 7, the LR model trained with all 10,000 TDPs had the best estimates of 2 model parameters, i.e., the intercept and the coefficient of VI, over the LR models trained with fewer TDPs. Meanwhile, the LR model trained with 1000 TDPs only had one best estimate, i.e., the coefficient of IM. The LR model trained with the fewest TDPs (500) had the most deviated estimates of the model parameters. Regarding the regression models, the MLR model trained with all 2809 TDPs had the best estimates of 3 model parameters, i.e., the intercept and the coefficients of IM and VI. Although the MLR model trained with 500 TDPs had the most deviated estimates of these 3 model parameters than the ones of the MLR model trained with 250 TDPs, the 500-TDP MLR model had the best estimate of σ_{MLR} among all 3 calibrated models. In general, the number of TDPs seemed to have an impact on the model calibration. This result was consistent with the intuition that the fewer TDPs, the more likely and to a larger extent a trained LR or MLR model would have estimates deviating from the values of parameters of the true model. Despite this impact, as few as hundreds of TDPs could still result in good estimates of parameters of an LR or MLR model for estimating societal vulnerability, as all the parametric estimates derived in this sensitivity analysis listed in Table 7 were close to the values of parameters of the true models.

The sensitivity analysis for Scenario 1V was also conducted to examine the effect of number of TDPs on the predictive performance of the individual ML-EPMs. As shown in Fig. 13(a)–(d), the classification ML-EPMs trained with 1000 and 500 TDPs produced ROC curves almost identical to the one of the true model in this run. As listed in Table 3, the classification ML-EPMs trained with fewer TDPs also resulted in validation statistics comparable to the ones of the models trained with all 10,000 TDPs and the true model. Despite the good performances of the classification ML-EPMs with fewer TDPs in this run, however, such good performances were not guaranteed. In some of the other attempted runs, the RFC and SVC models with the fewer TDPs could produce significantly worse ROC curves and validation statistics than the models with all 10,000 TDPs. In some runs, the optimization algorithm for calibrating ANNC models even failed to converge with fewer TDPs. Although there were undesired issues with the calibrations of the RFC, SVC, and ANNC models with fewer TDPs during some runs, the LR model always produced good ROC curves and validation statistics with fewer TDPs.

As per regression for Scenario 1V, the degree of negative effect of reducing number of TDPs on the predictive performance of an ML-EPM varies across the considered models. As shown in Fig. 13(e)–(h) and Table 4, the MLR model was the least affected by reduction of TDPs,

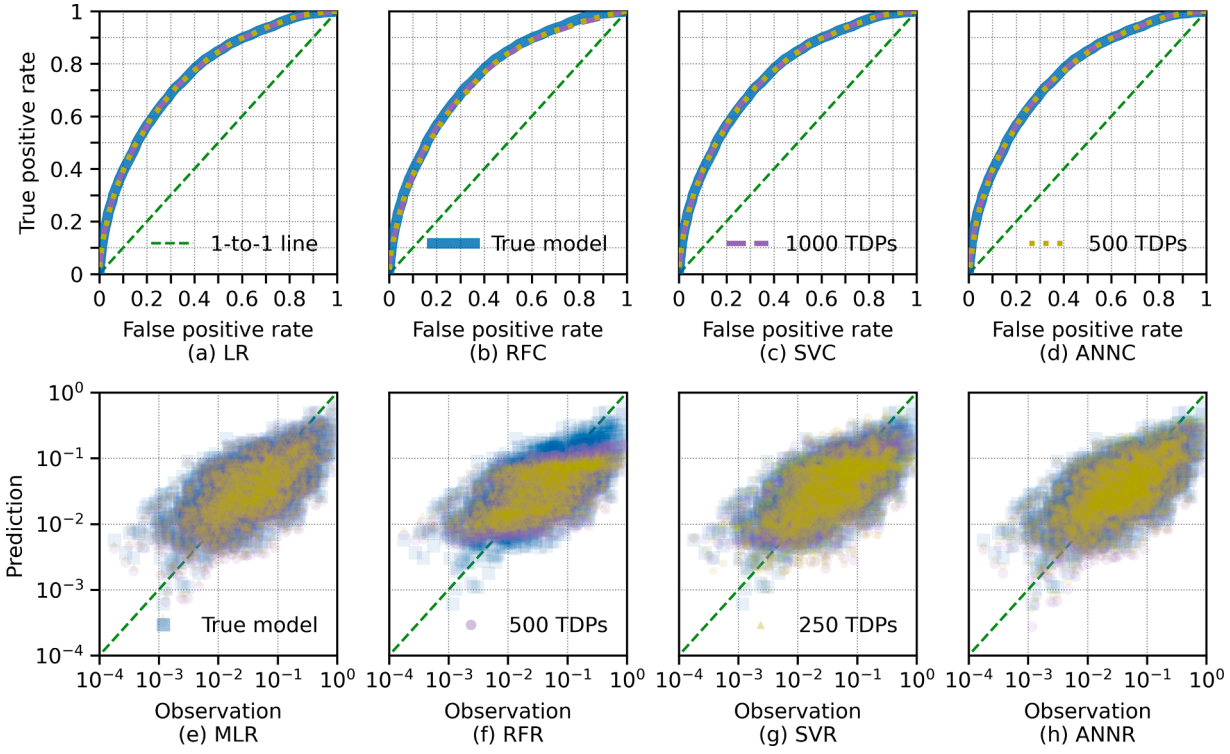


Fig. 13. Predictive performances as the result of sensitivity analysis in Scenario 1V in terms of the ROC curves produced by the (a) LR, (b) RFC, (c) SVC, and (d) ANNC models trained with 1000 and 500 TDPs and the distributions of data points predicted by the (e) MLR, (f) RFR, (g) SVR, and (h) ANNR models trained with 500 and 250 TDPs.

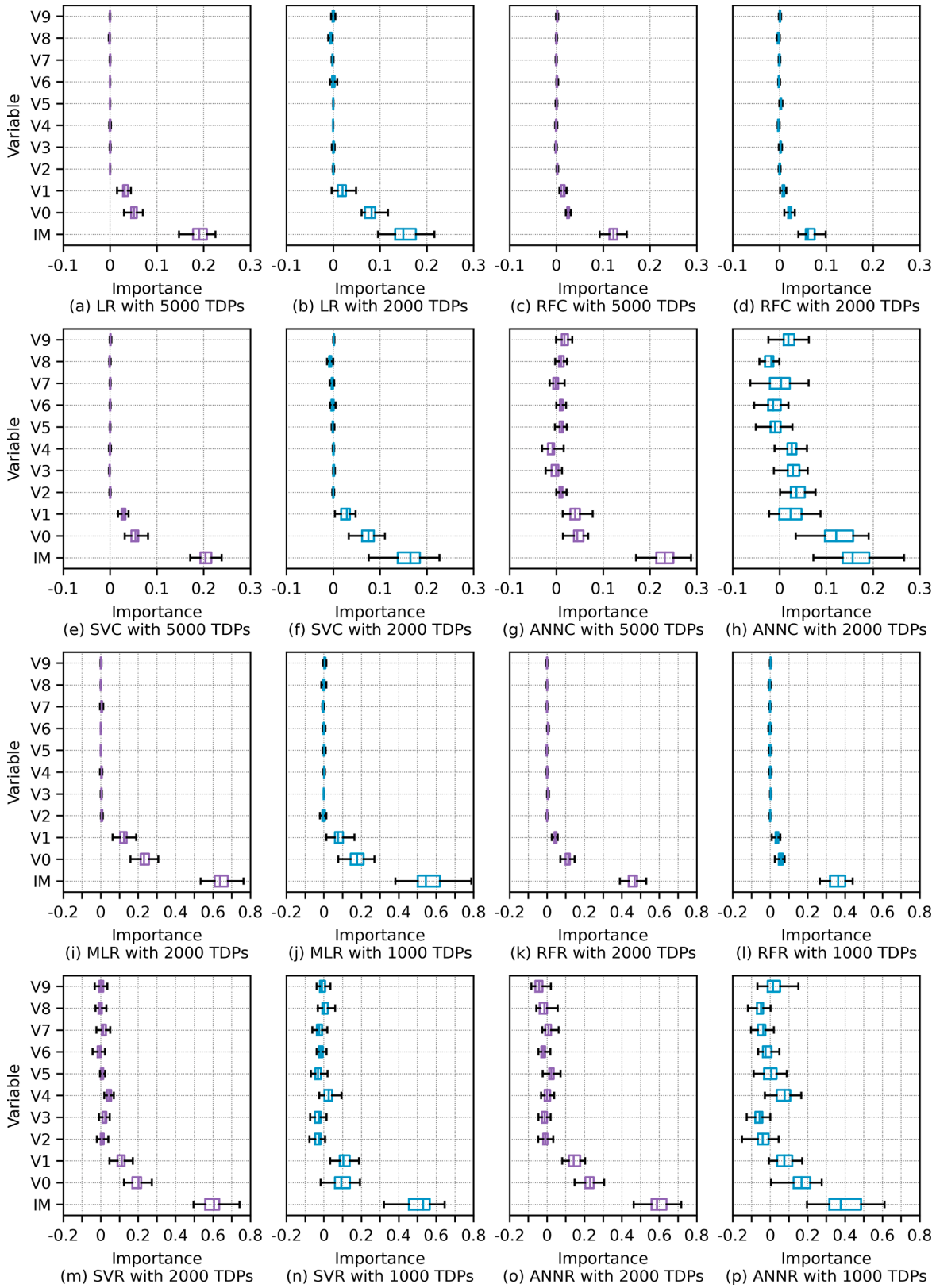


Fig. 14. Result of sensitivity analysis in terms of estimated importance measures of input variables of (a) the LR model trained with 5000 TDPs, (b) the LR model trained with 2000 TDPs, (c) the RFC model trained with 5000 TDPs, (d) the RFC model trained with 2000 TDPs, (e) the SVC model trained with 5000 TDPs, (f) the SVC model trained with 2000 TDPs, (g) the ANNC model trained with 5000 TDPs, (h) the ANNC model trained with 2000 TDPs, (i) the MLR model trained with 2000 TDPs, (j) the MLR model trained with 1000 TDPs, (k) the RFR model trained with 2000 TDPs, (l) the RFR model trained with 1000 TDPs, (m) the SVR model trained with 2000 TDPs, (n) the SVR model trained with 1000 TDPs, (o) the ANNR model trained with 2000 TDPs, and (p) the ANNR model trained with 1000 TDPs in Scenario nV.

Table 8
Result of sensitivity analysis in terms of selections of variables in Scenario nV.

Part	Model	TDPs	IM	V0	V1
Classification	True	N/A	Yes	Yes	Yes
	LR	10,000	Yes	Yes	Yes
		5000	Yes	Yes	Yes
		2000	Yes	Yes	**No**
	RFC	10,000	Yes	Yes	Yes
		5000	Yes	Yes	Yes
		2000	Yes	Yes	**No**
	SVC	10,000	Yes	Yes	Yes
		5000	Yes	Yes	Yes
		2000	Yes	Yes	**No**
	ANNC	10,000	Yes	Yes	Yes
		5000	Yes	Yes	**No**
		2000	Yes	Yes	**No**
Regression	True	N/A	Yes	Yes	Yes
	MLR	2836	Yes	Yes	Yes
		2000	Yes	Yes	Yes
		1000	Yes	Yes	**No**
	RFR	2836	Yes	Yes	Yes
		2000	Yes	Yes	Yes
		1000	Yes	Yes	**No**
	SVR	2836	Yes	Yes	Yes
		2000	Yes	Yes	Yes
		1000	Yes	**No**	**No**
	ANNR	2836	Yes	Yes	Yes
		2000	Yes	Yes	Yes
		1000	Yes	**No**	**No**

Note: **Bold** indicates mis-selection; the other variables including V2–V9 were not selected by any model.

while the RFR model was the most affected. Fig. 13(f) shows a significant drop in predictive performance of the RFR model when it was trained with 500 and 250 TDPs. Meanwhile, the SVR models trained with fewer TDPs also tended to produce poorer predictive performances than the SVR models trained with all 2809 TDPs as listed in Table 4 and shown in Fig. 13(g). Although the predictive performances of the ANNR model trained with fewer TDPs looked relatively good in this run, the ANNR model may not be trained properly with fewer TDPs in some attempted runs that resulted in exceptionally poor predictive performances. These results for Scenario 1V indicated that we needed at least hundreds of TDPs to properly train and calibrate ML-EPMs for quantifying societal vulnerability.

With Scenario nV, we conducted a sensitivity analysis to examine the effect of reducing the number of TDPs of ML-EPMs on their variable selections. As shown in Fig. 14 and Table 8, the reduction of number of TDPs had a negative impact on variable selection of all examined ML-EPMs. With 2000 and 1000 TDPs for classification and regression respectively, all examined ML-EPMs missed at least V1 when performing variable selection (Table 8). Among the ML-EPMs, the ANNC and ANNR models seemed to be the most affected by the reduction of number of TDPs in variable selection, as the uncertainties of the importance measures of input variables of the ANNC and ANNR models became more significant with fewer TDPs (Fig. 14(g), (h), (o), and (p)). This result indicated that, to guarantee a successful selection of pertinent input variables, at least thousands of TDPs were needed to train an ML-EPM. In particular, the ANN-based methods may need more TDPs than the other ML methods for variable selection. In real-world applications, however, it can be challenging to collect as many as thousands of TDPs with good distributions of values of IM and VIs to select pertinent variables to model societal vulnerability. This may require researchers to spend a long time waiting for many hazardous events to occur before obtaining a sufficient number of TDPs for modeling. In addition, real-world data on VIs are likely to face other inconsistency problems such as missing values and differing resolutions. Future work needs to develop methods to simulate data points with inconsistency issues to further examine the sensitivity of ML-EPMs, in terms of predictive performance, to the number and quality of TDPs for modeling societal vulnerability.

5. Conclusion

In this study, we proposed an MC approach to generate data to experimentally verify EPMs of societal vulnerability to losses associated with hazardous events. To model the correlational structure of data on IM and VIs, we adopted a novel eigenvalue-based method for data simulation. To examine EPMs, we designed true models based on LR and MLR modeling to simulate loss data of societal systems. As a demonstration, 8 commonly used ML-EPMs were tested and shown to be capable of selecting pertinent input variables with the PIA and producing predictive performances that were comparable to the ones of the true models. Results of the study also suggested that simple models should be preferred for creating vulnerability curves, as the extrapolation of vulnerability curves for large IMs with the sophisticated ML models may not be reliable. To successfully select the pertinent input variables and calibrate ML-EPMs for quantifying societal vulnerability, at least hundreds of TDPs with information on event IM, pre-event VIs, and event losses should be collected for model training, especially for training nonlinear models. The proposed MC methodology can also be extended to examine the veracity and validity of other methods for quantifying societal vulnerability, including those for social vulnerability. Future work needs to explore more advanced methods to model the correlational structure of data on IM and VIs for simulations of input variables of models of societal vulnerability. More true models that are more sophisticated than the LR-MLR model should be properly designed for simulation of loss data and for comparisons of model performances. More methods for variable selection should also be tested with the proposed experimental methodology in future studies.

Author statement

The authors declare that they have NOT used any generative artificial intelligence (AI) or AI-assisted technologies during the writing process.

CRediT authorship contribution statement

Yi Victor Wang: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Project administration, Validation, Software, Visualization, Writing – review & editing, Writing – original draft. **Seung Hee Kim:** Conceptualization, Funding acquisition, Project administration, Resources, Supervision, Writing – review & editing. **Menas C. Kafatos:** Conceptualization, Funding acquisition, Project administration, Supervision, Writing – review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

The authors thank Alexis Yankopoulos for her help during the research.

References

[1] Cumberbatch J, Drakes C, Mackey T, Nagdee M, Wood J, Degia AK, Hinds C. Social vulnerability index: Barbados—A case study. Coast Manag 2020;48(5):505–26.
[2] Cutter SL, Boruff BJ, Shirley WL. Social vulnerability to environmental hazards. Social Sci Q 2003;84(2):242–61.

[3] Flanagan BE, Gregory EW, Hallisey EJ, Heitgerd JL, Lewis B. A social vulnerability index for disaster management. J Homel Secur Emerg Manag 2011;8(1). https://doi.org/10.2202/1547-7355.1792.
[4] Holand IS, Lujala P, Rød JK. Social vulnerability assessment for Norway: a quantitative approach. Norwegian J Geogr 2011;65(1):1–17.
[5] Martins VN, e Silva DS, Cabral P. Social vulnerability assessment to seismic risk using multicriteria analysis: the case study of Vila Franca do Campo (São Miguel Island, Azores, Portugal). Nat Hazards 2012;62(2):385–404.
[6] Tate E. Uncertainty analysis for a social vulnerability index. Ann Assoc Am Geogr 2013;103(3):526–43.
[7] Wang S, Zhang M, Huang X, Hu T, Sun QC, Corcoran J, Liu Y. Urban–rural disparity of social vulnerability to natural hazards in Australia. Sci Rep 2022;12(1):1–5.
[8] Yoon DK. Assessment of social vulnerability to natural disasters: a comparative study. Nat Hazards 2012;63(2):823–43.
[9] Bakkensen LA, Fox-Lent C, Read LK, Linkov I. Validating resilience and vulnerability indices in the context of natural disasters. Risk Anal 2017;37(5):982–1004.
[10] Rufat S, Tate E, Emrich CT, Antolini F. How valid are social vulnerability models? Ann Am Assoc Geogr 2019;109(4):1131–53.
[11] Wang YV, Gardoni P, Murphy C, Guerrier S. Empirical predictive modeling approach to quantifying social vulnerability to natural hazards. Ann Assoc Am Geogr 2021;111(5):1559–83.
[12] Arrighi C, Mazzanti B, Pistone F, Castelli F. Empirical flash flood vulnerability functions for residential buildings. SN Appl Sci 2020;2(5):1–12.
[13] Baker JW. Efficient analytical fragility function fitting using dynamic structural analysis. Earthq Spectra 2015;31(1):579–99.
[14] Gardoni P, Mosalam KM, Der Kiureghian A. Probabilistic seismic demand models and fragility estimates for RC bridges. J Earthq Eng 2003;7(spec01):79–106.
[15] Mardfekri M, Gardoni P. Probabilistic demand models and fragility estimates for offshore wind turbine support structures. Eng Struct 2013;52:478–87.
[16] Nofal OM, van de Lindt JW, Do TQ. Multi-variate and single-variable flood fragility and loss approaches for buildings. Reliab Eng Syst Saf 2020;202:106971.
[17] Papathoma-Köhle M, Keiler M, Totschnig R, Glade T. Improvement of vulnerability curves using data from extreme events: debris flow event in South Tyrol. Nat Hazards 2012;64(3):2083–105.
[18] Tavares DH, Padgett JE, Paultre P. Fragility curves of typical as-built highway bridges in eastern Canada. Eng Struct 2012;40:107–18.
[19] Xu H, Gardoni P. Probabilistic capacity and seismic demand models and fragility estimates for reinforced concrete buildings based on three-dimensional analyses. Eng Struct 2016;112:200–14.
[20] Lallemant D, Kiremidjian A, Burton H. Statistical procedures for developing earthquake damage fragility curves. Earthq Eng Struct Dyn 2015;44(9):1373–89.
[21] Argyroudis SA, Mitoulis SA. Vulnerability of bridges to individual and multiple hazards—Floods and earthquakes. Reliab Eng Syst Saf 2021;210:107564.
[22] Vishwanath BS, Banerjee S. Considering uncertainty in corrosion process to estimate life-cycle seismic vulnerability and risk of aging bridge piers. Reliab Eng Syst Saf 2023;232:109050.
[23] Ding L, Khan F, Ji J. A novel vulnerability model considering synergistic effect of fire and overpressure in chemical processing facilities. Reliab Eng Syst Saf 2022;217:108081.
[24] Gangolu J, Kumar A, Bhuyan K, Sharma H. Probabilistic demand models and performance-based fragility estimates for concrete protective structures subjected to missile impact. Reliab Eng Syst Saf 2022;223:108497.
[25] Vrogushyn S, Merz B, Apel H. Development of dike fragility curves for piping and micro-instability breach mechanisms. Nat Hazards Earth Syst Sci 2009;9(4):1381–401.
[26] Caratozzolo V, Misuri A, Cozzani V. A generalized equipment vulnerability model for the quantitative risk assessment of horizontal vessels involved in Natech scenarios triggered by floods. Reliab Eng Syst Saf 2022;223:108504.
[27] Rossi L, Moreno VC, Landucci G. Vulnerability assessment of process pipelines affected by flood events. Reliab Eng Syst Saf 2022;219:108261.
[28] Lanzano G, Salzano E, de Magistris FS, Fabbrocino G. Seismic vulnerability of natural gas pipelines. Reliab Eng Syst Saf 2013;117:73–80.
[29] Jin S, Gong J. Fragility analysis and probabilistic performance evaluation of nuclear containment structure subjected to internal pressure. Reliab Eng Syst Saf 2021;208:107400.
[30] Zhao C, Yu N, Peng T. Probabilistic seismic fragility assessment of isolated nuclear power plant structure using IDA and MSA methods. Structures 2021;34:1300–11.
[31] Zhao Y-G, Qin M-J, Lu Z-H, Zhang L-W. Seismic fragility analysis of nuclear power plants considering structural parameter uncertainty. Reliab Eng Syst Saf 2021;216:107970.
[32] Dikshit S, Alipour A. A moment-matching method for fragility analysis of transmission towers under straight line winds. Reliab Eng Syst Saf 2023;236:109241.
[33] Ma L, Bocchini P, Christou V. Fragility models of electrical conductors in power transmission networks subjected to hurricanes. Struct Saf 2020;82:101890.
[34] Zhang J, Bagtzoglou Y, Zhu J, Li B, Zhang W. Fragility-based system performance assessment of critical power infrastructure. Reliab Eng Syst Saf 2023;232:109065.
[35] Bellè A, Zeng Z, Duval C, Sango M, Barros A. Modeling and vulnerability analysis of interdependent railway and power networks: application to British test systems. Reliab Eng Syst Saf 2022;217:108091.
[36] Kishore KB, Gangolu J, Ramancha MK, Bhuyan K, Sharma H. Performance-based probabilistic deflection capacity models and fragility estimation for reinforced concrete column and beam subjected to blast loading. Reliab Eng Syst Saf 2022;227:108729.
[37] Ceferino L, Lin N, Xi D. Bayesian updating of solar panel fragility curves and implications of higher panel strength for solar generation resilience. Reliab Eng Syst Saf 2023;229:108896.
[38] Li X, Chen G, Amyotte P, Khan F, Alauddin M. Vulnerability assessment of storage tanks exposed to simultaneous fire and explosion hazards. Reliab Eng Syst Saf 2023;230:108960.
[39] Yang Y, Chen G, Reniers G. Vulnerability assessment of atmospheric storage tanks to floods based on logistic regression. Reliab Eng Syst Saf 2020;196:106721.
[40] FEMA (US Federal Emergency Management Agency) (2022). Hazus User and Technical Manuals. Accessed September 21, 2022: https://www.fema.gov/flood-maps/tools-resources/flood-map-products/hazus/user-technical-manuals.
[41] MAE Center (Creating a Multi-Hazard Approach to Engineering Center). (2022) MAEViz. Accessed September 21, 2022: http://mae.cee.illinois.edu/software/software_maeviz.html.
[42] Wang YV, Gardoni P, Murphy C, Guerrier S. Predicting fatality rates due to earthquakes accounting for community vulnerability. Earthq Spectra 2019;35(2):513–36.
[43] Wang YV, Gardoni P, Murphy C, Guerrier S. Worldwide predictions of earthquake casualty rates with seismic intensity measure and socioeconomic data: a fragility-based formulation. Nat Hazards Rev 2020;21(2):04020001.
[44] Chen W, Zhang L. An automated machine learning approach for earthquake casualty rate and economic loss prediction. Reliab Eng Syst Saf 2022;225:108645.
[45] Firuzi E, Hosseini KA, Ansari A, Izadkhah YO, Rashidabadi M, Hosseini M. An empirical model for fatality estimation of earthquakes in Iran. Nat Hazards 2020;103:231–50.
[46] Jaiswal K, Wald D. An empirical model for global earthquake fatality estimation. Earthq Spectra 2010;26(4):1017–37.
[47] Wu S, Jin J, Pan T. Empirical seismic vulnerability curve for mortality: case study of China. Nat Hazards 2015;77(2):645–62.
[48] Wang YV, Sebastian A. Community flood vulnerability and risk assessment: an empirical predictive modeling approach. J Flood Risk Manag 2021;14(3):e12739.
[49] Python Software Foundation. (2022). Python 3.8.11. Accessed September 21, 2022: https://www.python.org/.
[50] The World Bank. (2022). World Development Indicators. Accessed September 21, 2022: https://datacatalog.worldbank.org/search/dataset/0037712.
[51] Smirnov N. Table for estimating the goodness of fit of empirical distributions. Ann Math Stat 1948;19(2):279–81.
[52] Scikit-learn. (2022). Scikit-learn: machine learning in Python. Accessed September 21, 2022: https://scikit-learn.org/stable/.
[53] Fukushima K. Visual feature extraction by a multilayered network of analog threshold elements. IEEE Trans Syst Sci Cybern 1969;5(4):322–33.
[54] Glorot X, Bordes A, Bengio Y. Deep sparse rectifier neural networks. Proc Mach Learn Res 2011;15:315–23.
[55] TensorFlow. (2022). Create Production-Grade Machine Learning Models with TensorFlow. Accessed September 21, 2022: https://www.tensorflow.org/.
[56] Guyon I, Elisseeff A. An introduction to variable and feature selection. J Mach Learn Res 2003;3:1157–82.
[57] Tibshirani R. Regression shrinkage and selection via the Lasso. J R Stat Soc 1996;58(1):267–88.
[58] Wang YV, Kim SH, Lyu G, Lee C-L, Lee G, Min K-H, Kafatos MC. Relative importance of radar variables for nowcasting heavy rainfall: a machine learning approach. IEEE Trans Geosci Remote Sens 2023;61:4100314.
[59] Altmann A, Toloşi L, Sander O, Lengauer T. Permutation importance: a corrected feature importance measure. Bioinformatics 2010;26(10):1340–7.
[60] Wang YV, Sebastian A. Murphy Scale: A locational equivalent intensity scale for hazard events. Risk Analysis 2023;43(3):605–23.