Scientific Research Publishing

Journal of Data Analysis and Information Processing, 2024, 12, 319-347
https://www.scirp.org/journal/jdaip
ISSN Online: 2327-7203
ISSN Print: 2327-7211

Spatial Heterogeneity Modeling Using Machine Learning Based on a Hybrid of Random Forest and Convolutional Neural Network (CNN)

Amadou Kindy Barry[1*], Anthony Waititu Gichuhi[2], Lawrence Nderu[3]

[1]Department of Mathematics, Pan African University, Institute for Basic Sciences, Technology and Innovation (PAUISTI), Nairobi, Kenya
[2]Department of Statistics and Actuarial Science, Jomo Kenyatta University of Agriculture and Technology, Nairobi, Kenya
[3]Department of Computing School and Information Technology (SCIT), Jomo Kenyatta University of Agriculture and Technology (JKUAT), Nairobi, Kenya
Email: *kindy.amadou@students.jkuat.ac.ke

How to cite this paper: Barry, A.K., Gichuhi, A.W. and Nderu, L. (2024) Spatial Heterogeneity Modeling Using Machine Learning Based on a Hybrid of Random Forest and Convolutional Neural Network (CNN). *Journal of Data Analysis and Information Processing*, **12**, 319-347.
https://doi.org/10.4236/jdaip.2024.123018

Received: April 24, 2024
Accepted: June 10, 2024
Published: June 13, 2024

Abstract

Spatial heterogeneity refers to the variation or differences in characteristics or features across different locations or areas in space. Spatial data refers to information that explicitly or indirectly belongs to a particular geographic region or location, also known as geo-spatial data or geographic information. Focusing on spatial heterogeneity, we present a hybrid machine learning model combining two competitive algorithms: the Random Forest Regressor and CNN. The model is fine-tuned using cross validation for hyper-parameter adjustment and performance evaluation, ensuring robustness and generalization. Our approach integrates Global Moran's I for examining global autocorrelation, and local Moran's I for assessing local spatial autocorrelation in the residuals. To validate our approach, we implemented the hybrid model on a real-world dataset and compared its performance with that of the traditional machine learning models. Results indicate superior performance with an R-squared of 0.90, outperforming RF 0.84 and CNN 0.74. This study contributed to a detailed understanding of spatial variations in data considering the geographical information (Longitude & Latitude) present in the dataset. Our results, also assessed using the Root Mean Squared Error (RMSE), indicated that the hybrid yielded lower errors, showing a deviation of 53.65% from the RF model and 63.24% from the CNN model. Additionally, the global Moran's I index was observed to be 0.10. This study underscores that the hybrid was able to predict correctly the house prices both in clusters and in dispersed areas.

Keywords

Spatial Heterogeneity, Spatial Data, Feature Selection, Standardization, Machine Learning Models, Hybrid Models

1. Introduction

In recent years, there has been an exponential growth in data generation, with a significant portion constituting geospatial data, which encompasses various sources like remote sensing imagery, GPS trajectories, and weather observations. Big data, including geospatial data, is characterized by its volume, velocity, and variety, presenting opportunities to explore real-time insights. Geospatial big data, with its substantial volume and potential for real-time updates, offers new avenues for uncovering insights about our environment [1]. Spatial data refers to information that explicitly or indirectly pertains to a particular geographic region or location, also known as geo-spatial data or geographic information. To fully understand spatial data, machine learning and spatial statistics need to work together. Researchers are developing ways to combine these techniques for a more comprehensive approach to modeling spatial phenomena [2]. Spatial attributes are utilized to specify the spatial position and scope of spatial entities. These attributes commonly include details regarding spatial coordinates, such as longitude, latitude, and elevation [3].

One of the properties of spatial data that offer the most encouraging prospects for the future of spatial machine learning is spatial heterogeneity. Spatial heterogeneity can be defined as the presence of variation in the relationships between dependent and independent variables across the space. According to [4], spatial heterogeneity is either defined as the difference in space in distribution of a point pattern, or difference of a qualitative or quantitative value of a surface pattern. Spatial heterogeneity, or geographic variation [5], matters in machine learning models. Ignoring it can lead to inaccurate predictions in some areas. By accounting for these local patterns, models can improve their overall performance and become more generalizable, meaning they work well on new unseen data from the same region.

Machine learning (ML) is a subset of AI that teaches machines about how to imitate the intelligence of human behavior. The four main approaches of ML are: supervised learning, unsupervised learning, semi-supervised learning and reinforcement learning. Given the rise of extensive geospatial datasets, machine learning (ML) has become widely integrated across various domains within geoscience research. This includes applications in tasks such as land cover classification [6], assessment of landslide susceptibility [7], investigations into climate change impacts, and analyses of atmospheric dynamics [8] [9].

Spatial methods are rooted in the principle of spatial heterogeneity, acknowledging the diverse variations across geographical entities. This spatial hetero-

geneity introduces complex patterns and distributions across space, challenging the assumption of uniformity and independence in traditional statistical methods. While the first law of geography highlights spatial dependence, spatial heterogeneity underscores the need to account for the varying characteristics and relationships among proximate entities, thereby shaping the spatial autocorrelation (SAC) observed in the dataset.

Spatial prediction, a primary application of machine learning with geospatial data, involves creating a model based on training data to predict values at particular locations where information is lacking [10]. Machine Learning algorithms are being explored as potential alternatives for conventional geostatistical interpolation methods (e.g. Ordinary Kriging, Universal kriging, Regression Kriging…) and spatial analysis techniques due to their advancements in computational power, data availability, and algorithmic innovations [11]. Recent studies by authors [12] and [13] have demonstrated the advantage of machine learning models over geostatistical interpolation techniques in prediction performance. In [14], they introduced a framework called RFsp, which utilizes random forest for spatial predictions by incorporating buffer distances from observation points as explanatory variables. [15] used systematic approach for determining the appropriate range of scales, including upper and lower limits, for spatial modeling using machine learning techniques. The method is designed to enhance modeling accuracy and is evaluated for its effectiveness in achieving this objective. Geographical Random Forest (GRF) serves as a valuable exploratory tool for visualizing the relationship between dependent and independent variables, thereby elucidating local variations and enhancing comprehension of the underlying processes contributing to observed spatial heterogeneity [16]. [17] used the machine learning algorithm Random Forests to train models using non-spatial and spatial cross-validation strategies to understand how spatial variable selection affects the predictions. In machine learning [18], modeling with geographic coordinates or Euclidean distance fields can produce similar interpolations resembling linear variograms with infinite ranges.

Current research exploring the fusion of machine learning and spatial analysis remains relatively scarce or limited. Hybrid methodologies, notably combining Geographically Weighted Regression (GWR) from geostatistics and Artificial Neural Networks (ANN) from machine learning, have incorporated the estimation of residuals using ordinary kriging [19]. Both GWRK (geographically weighted regression kriging) and ANNK (artificial neural networks kriging) hybrid models are capable of integrating the spatial autocorrelation of observed variables, leading to enhanced predictive performance and reduced errors. [20] used an approach called Euclidean distance fields in machine-learning (EDM). This method provides advantages over other prediction methods that integrate spatial dependence and state factor models, for example, regression kriging (RK) and geographically weighted regression (GWR). The use of the geographically-weighted random forest (GW-RF) model to understand the spatial relationship between Type 2 Diabetes (T2D) prevalence and its risk factors, and how this va-

ries across different geographical areas, [21] is vital for identifying regions in need of targeted efforts and resources to alleviate the burden of T2D.

Moran's I is the most widely used measurements to account for spatial autocorrelation. To evaluate the presence of spatial dependencies within models, the Moran's I statistic was determined, and a novel weights matrix integration was developed to identify spatial dependency patterns within residuals [22]. The investigation of spatial heterogeneity within a machine learning algorithm is still in its initial phase. Spatial variations in data are driving the creation of deep machine learning models like CNNs, RNNs, … that can handle these geographical differences [23]. To ensure these models work well, researchers are developing ways to assess their performance while considering these variations [24]. This includes checking for spatial patterns in errors and ensuring the model works well in unseen location [25]. To the best of our Knowledge, no hybrid machine learning model based on RF and CNN has been developed to account for spatial heterogeneity using Global and Local Moran's I with the residuals of the models.

In the following research, we will develop a hybrid model based on Random Forest (RF) and Convolutional Neural Network (CNN) architectures specifically designed for spatial data analysis. This study focuses on exploiting the capabilities of a hybrid machine-learning model to capture spatial dependencies through the residuals without the need for explicit spatial features in the model. We propose a combination of RF, known for its effectiveness in capturing spatially dependent samples, and CNN, renowned for its ability to handle complex spatial relationships and patterns. These two representative machine learning models will be integrated into a hybrid framework designed for spatial heterogeneity.

The paper follows the subsequent structure: Section 2 mentions the literature review, section 3 outlines the mathematical methods employed for the models, Section 4 presents the experimental findings, Section 5 engages in a comprehensive discussion, and Section 6 details the conclusions drawn from this study.

2. Related Works

[26] presented a new method, geographically weighted Extreme Learning Machine (GWELM), to address spatial heterogeneity within data. By adapting the Extreme Learning Machine and incorporating spatially varying parameters estimated through geographically weighted least squares, GWELM outperforms comparative methods in capturing spatial variations across two diverse datasets. This highlights the method's efficacy in effectively addressing spatial heterogeneity.

Heterogeneous space-time artificial neural networks (HSTANNs) were developed [27] to improve space-time series prediction by integrating spatial and temporal dependencies as well as heterogeneity. The method involves clustering the study area into homogeneous sub-areas to handle spatial heterogeneity and analyzing space-time autocorrelation to understand the dataset's space-time dependence structure. Experimental results indicated that HSTANNs significantly enhance forecasting accuracy compared to alternative methods, underscoring their effectiveness in capturing complex spatiotemporal patterns.

[28] developed a hybrid model to enhance landslide susceptibility mapping by integrating GeoSOM and Stacking ensemble methods. The GeoSOM method was utilized to address spatial heterogeneity by clustering the study area into homogeneous regions, assigning each region a cluster attribute as a model input. Meanwhile, the Stacking ensemble technique combined support vector machine (SVM), artificial neural network (ANN), and gradient-boosting decision tree (GBDT) methods to create a high-performance landslide model. The results demonstrated that the hybrid model outperformed traditional machine learning methods, achieving an AUC score 0.11 - 0.135 higher than those of individual methods.

[29] proposed a deep stacking ensemble model to improve the prediction performance of heart disease. This ensemble integrates two optimized and pretrained hybrid deep learning models: Convolutional Neural Network (CNN)-Long Short-Term Memory (LSTM) (CNN-LSTM) and CNN-Gated Recurrent Unit (GRU). The Support Vector Machine (SVM) serves as the meta-learner model. Comparative analysis against five machine learning models and hybrid models reveals that the proposed ensemble achieves the highest performance, especially when using the full feature set.

The study introduces GWANN [30], a geographically weighted artificial neural network, to derive urban CA transition rules, considering spatial heterogeneity and nonlinearity. Examining urban sprawl in Wuhan and Beijing from 2000 to 2020, GWANN's effectiveness is compared with LR, GWLR, and ANN. Results indicate GWANN's superior fitting and simulation performance, emphasizing the importance of integrating spatial heterogeneity and nonlinearity for accurate transition rule establishment in urban sprawl modeling.

To examine the impact of lockdowns on the Air Quality Index (AQI), [31] utilized a deep learning framework, incorporating spatial autocorrelation (SAC), which integrates temporal and spatial correlation, the analysis estimates lockdown effects of −25.88 in Wuhan and −20.47 in Shanghai. These predictions significantly reduce prediction errors by around 47% for Wuhan and 67% for Shanghai, improving the reliability of AQI forecasts in both cities.

Less research exists on the integration of machine learning and spatial analysis, but it presents some limitations and opportunities for further research. The integration of RF and CNN machine learning algorithms with spatial analysis techniques represents a novel approach to spatial heterogeneity modeling. This integration aims to capture the spatial patterns present in the data by leveraging the strengths of both models.

3. Material and Methods

3.1. Dataset and Tools

The study utilizes a publicly available spatial dataset sourced from Kaggle, focusing on California housing prices. Originally used by Dr. Kelley Pace and Dr. Ronald Barry, this dataset served as the foundation for constructing spatial au-

to-regressive models based on 1990 California Census data. It is commonly used for spatial autocorrelation analysis: https://www.kaggle.com/datasets/camnugent/california-housing-prices. And it encompasses details regarding district demographics such as income, population, and households, alongside spatial coordinates (latitude, longitude), and a comprehensive description of each district's residential properties including the number of rooms, bedrooms, house value, and proximity to the ocean. With a substantial dataset comprising 20,640 observations of housing prices and 10 features, in those 10 features, 9 features represent input features and the feature *median_house_value* is the target/response variable, each observation represents a distinct block in California. The attributes of the dataset are elaborated in **Table 1**, demonstrating a sample of the dataset. Furthermore, **Figure 1** illustrates the distribution of each variable, providing insights into their spread. Meanwhile, **Figure 2** illustrates the dispersion of median home values across California concerning both population density and geographical location. Notably, it reveals a clear trend where houses closer to the ocean tend to exhibit higher median values. It is common for houses situated near bodies of water or in high population density to command higher prices compared to those located further inland. Consequently, incorporating spatial information becomes essential in accurately predicting housing prices.

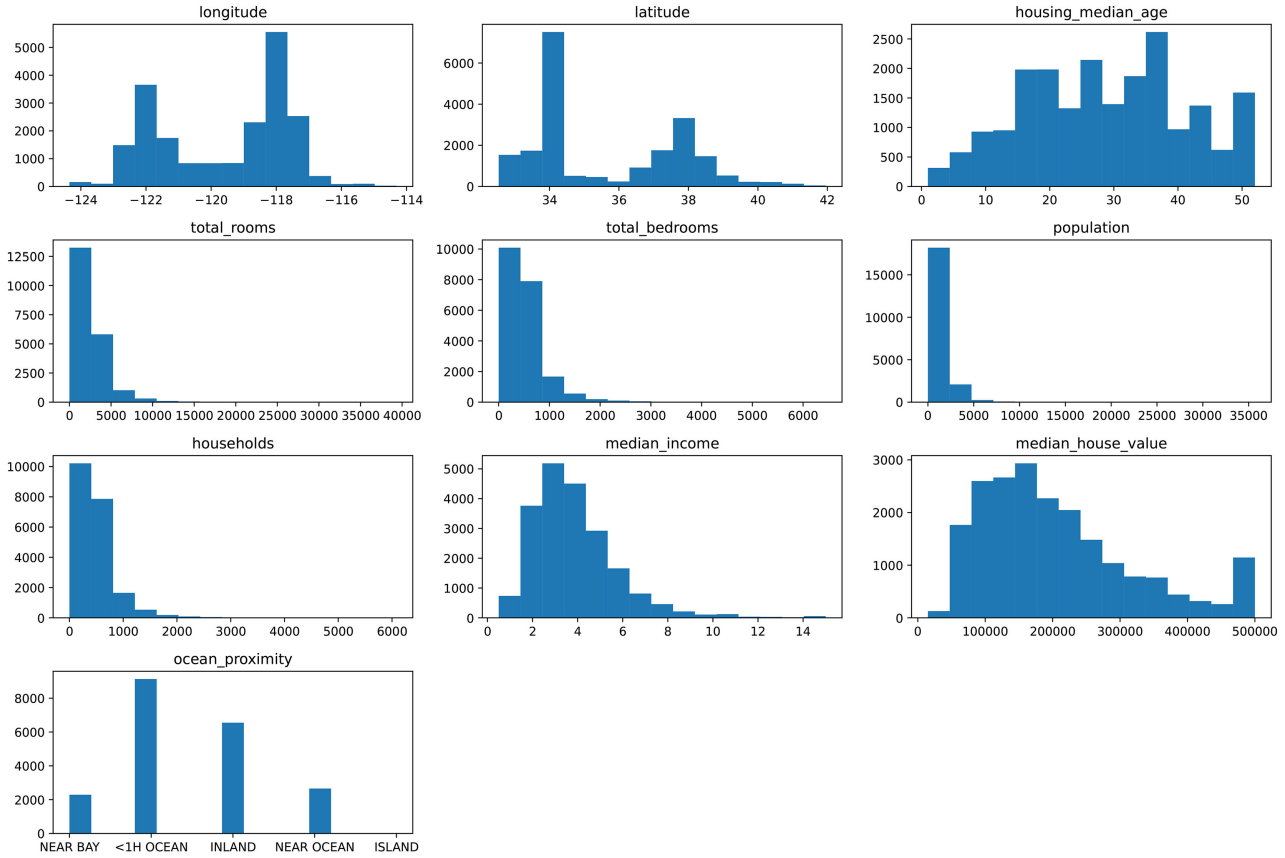

Figure 1. Data distribution.

Table 1. Description of variables for California housing prices.

Variable	Definition
longitude	Longitude of the house.
latitude	Latitude of the house
housing_median_age	Median age of a house within a bloc
total_rooms	Total number of rooms within a housing bloc
total_bedrooms	Total number of bedrooms within a housing bloc
population	Total number of people living within a bloc
households	Total number of households within a bloc
median_income	Median income for households within a block of houses
median_house_value	Median house value for households within a block
ocean_proximity	Location of the house in relation to the ocean

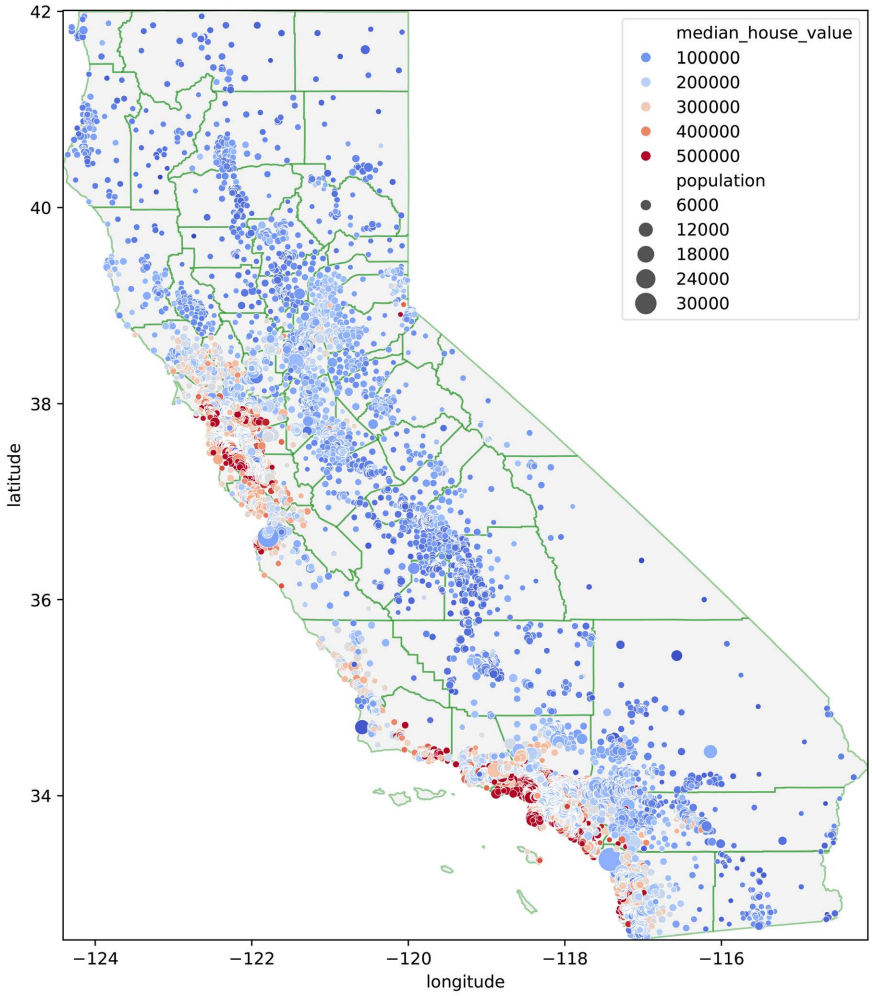

Figure 2. The spread of housing prices among the population in California.

For the accomplishment of this work the **Python** 3.11.7 open-source software was utilized, along with various powerful Package and libraries like Scikit-learn 1.3.2, Pandas, Numpy, Matplotlib, Scipy 1.11.4, TensorFlow 2.15.0 and Keras for

the Convolutional Neural Network, and geopandas 0.14.2 and PySal 24.01 for the spatial operations and analysis. This combination of tools and libraries allowed for a robust and comprehensive approach to data analysis, modeling and spatial analysis in the context of this work.

From the above histograms and bar graph of the different features, we can observe that certain features exhibit a skewness towards higher values, with the Median House Value being notably concentrated towards the higher end of the range. This suggest that there may be a prevalence of higher house values within the dataset, with fewer observations having lower values, and features are distributed on very different scales.

3.2. Data Preprocessing

Data preprocessing plays a crucial role in the data mining process as it involves tasks such as cleaning, transforming, and integrating data to prepare it for analysis. Real-world data often presents challenges such as irregular formats, missing values, outliers, errors, noise, etc. These issues may arise from factors such as human error during data collection, limitations of measurement tools, system malfunctions, or inherent variability in the real-world phenomena. The spatial dataset contains 20,640 instances for different districts in California and 10 attributes which are: longitude, latitude, housing_median_age, total_rooms, total_bedrooms, population, households, median_income, median_house_value and ocean_proximity. The "ocean_proximity" attribute is categorical which we converted to numeric type using one-hot encoding method due to the small number of categories. The variable "total_bedrooms" contains some missing values which were replaced with the median value of the variable. This preprocessing step was necessary because many machine learning models cannot handle datasets with missing values. The target attribute is the "median_house_value" which ranges from 14,999 to 500,001 expressed in US dollars.

3.2.1. Standardization

Standardization, often referred to as z-score normalization, is a statistical method utilized to adjust a dataset's distribution such that it possesses a mean of 0 and a standard deviation of 1. This alteration facilitates the comparison of data points across various scales, proving especially beneficial in both machine learning and statistical investigations. Mathematically, it can be written as:

$$x_s = \frac{x - \mu}{\sigma} \tag{1}$$

where x_s is the standardized value, x is the original value, μ is the mean of the dataset and σ is the standard deviation of the dataset.

In this study, the standardization process was implemented using the **StandardScaler** method from the scikit-learn library.

3.2.2. Data Training and Testing Process

In machine learning (ML), training and testing are crucial stages that enable al-

gorithms to discover insights from available data, generate predictions, and refine their performance gradually. The spatial dataset was splitted into a 70/30 ratio, 70% used to train the different models and 30% used to evaluate the generalization and capabilities of the models on unseen data. This is illustrated in Table 2.

3.3. Machine Learning Models

Machine Learning ranges a wide scope of research areas, encompassing a multitude of algorithms. This paper implements the following machine learning models: Random Forest (RF), Convolutional Neural Network (CNN) and the hybrid model.

3.3.1. Random Forest (RF)

In problems involving regression, Random Forest is a group of individual decision trees, each using a subset of features, that generally contribute their predictions to create a final output of the response variable through averaging [32]. Random Forest models can also capture non-linear relationships and complex interactions between input variables and the response variable. In spatial datasets, where connections maybe intricate and straightforward relationships are not present, the ability to capture complex patterns becomes particularly essential. In this study, the utilization of Random Forest (RF) is chosen due to its overall precision and proven effectiveness across a wide range of geoscientific challenges [33].

Let $X=\{x_1,x_2,\cdots,x_m\}$ be the inputs and $Y=\{y_1,y_2,\cdots,y_m\}$ be the corresponding outputs.

The mathematical expression of the Random Forest Regressor is as follows:

$$\hat{Y}_{rf}^{D}(x)=\frac{1}{D}\sum_{d=1}^{D}L_d(x) \tag{2}$$

where:

- $\hat{y}_{rf}^{D}(x)$ represents the output for input x using an ensemble of D decision trees.
- D is the number of decision trees in the Random Forest.
- $\sum_{d=1}^{D}$ denotes the sum running from $d=1$ to D.
- $L_d(x)$ is the output of the d^{th} decision tree for the input x.

3.3.2. Feature Selection

The integration of random forest regressors (RF) and convolutional neural networks (CNN) in this study involves feature selection. Feature selection allows for the identification of the best subset of features that are important for a given operation. Feature selection allows us to find the best subset of features that are

Table 2. The split description.

Initial dataset size	Training set	Testing set
20,640	14,448	6192

important for an operation. It also helps avoid over fitting and improve model performance. Furthermore, it enables us to achieve more profound understanding of the fundamental mechanisms that led to the data. The advantages of this integration include improved predictive performance, robustness to noise and outliers, and the ability to capture complex spatial patterns and relationships in the data.

In our research study, we opted for the Mean Decrease in Accuracy (MDA). MDA is a metric utilized in ensemble learning, notably in decision tree-based models like Random Forest, to assess the significance of individual features in prediction accuracy. It evaluates how each feature contributes to making accurate predictions. The MDA index relies on the permutation of out-of-bag (OOB) samples to determine the significance of a variable. OOB samples consist of observations that are not employed in constructing the current tree. They serve dual purpose of estimating prediction error and assessing the importance of variables [34].

Let F_{im} represents the feature importance score for each feature, X the original input dataset with features and $F_{im}(X)$ the vector of feature importance scores for all the features in X. For each feature x_i, calculate the Mean Decrease in Accuracy (MDA) as follows:

$$F_{im}(x_i) = \frac{1}{N}\sum_{t=1}^{N}\left(EP_{x_i}(t) - E_{x_i}(t)\right) \tag{3}$$

where:

- $F_{im}(x_i)$ is the feature importance score for feature x_i.
- N is the number of trees in the Random Forest.
- $E_{x_i}(t)$ represents out-of-bag error on tree t before permuting values of x_i.
- $EP_{x_i}(t)$ indicates out-of-bag error on tree t after permutation.

After the computation, we maintained features with non-zero F_{im} scores or those with the highest scores based on the difference between the score values.

Let X' represent the new dataset which contains only the features with non-zero importance scores. The mathematical expression is as follows:

$$X' = \left\{x_i \in X \mid F_{im}(x_i) \neq 0\right\} \tag{4}$$

where:

- X' represents the new dataset with features of non-zero importance scores.
- x_i is a feature in the original dataset X.
- $F_{im}(x_i)$ represents the importance score for feature x_i.

3.3.3. Convolutional Neural Network

A Convolutional Neural Network (CNN) tailored for spatial tabular data is a deep learning model designed to process structured datasets containing spatial or geographic information in tabular format. Unlike traditional CNNs, which primarily operate on grid-like structures such as images, a CNN for spatial tabular data may incorporate convolutional layers to extract spatial patterns and relationships from tabular datasets, while also leveraging fully connected layers to

process the tabular data [35]. Its normal structure is a stack of convolutional pooling layers followed by totally connected layers [36]. CNN is commonly formed of three types of layers (or components): convolution, pooling and fully connected layers. CNNs are particularly strong for extracting hierarchical spatial features. They can automatically learn and capture relevant patterns and relationships within the spatial structure of a dataset.

3.3.4. Converting the Matrix of Selected Features to a Tensor

Let X' with dimensions $n \times m$ be the selected-features matrix dataset:

$$X' = \begin{bmatrix} x'_{11} & x'_{12} & x'_{13} & \cdots & x'_{1m} \\ x'_{21} & x'_{22} & x'_{23} & \cdots & x'_{2m} \\ x'_{31} & x'_{32} & x'_{33} & \cdots & x'_{3m} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x'_{n1} & x'_{n2} & x'_{n3} & \cdots & x'_{nm} \end{bmatrix} \tag{5}$$

where:

- n represents the number of rows.
- m represents the number of columns.

In order to convert X' into a 3D tensor, we can include an additional dimension of size 1 which represents the number of channels since we are working with tabular data in this case. So, the resulting tensor will have dimensions of $n \times m \times 1$.

$$\underline{X} = \begin{bmatrix} \begin{bmatrix} x'_{11} \\ x'_{12} \\ x'_{13} \\ \vdots \\ x'_{1m} \end{bmatrix} \\ \begin{bmatrix} x'_{21} \\ x'_{22} \\ x'_{23} \\ \vdots \\ x'_{2m} \end{bmatrix} \\ \begin{bmatrix} x'_{31} \\ x'_{32} \\ x'_{33} \\ \vdots \\ x'_{3m} \end{bmatrix} \\ \vdots \\ \begin{bmatrix} x'_{n1} \\ x'_{n2} \\ x'_{n3} \\ \vdots \\ x'_{nm} \end{bmatrix} \end{bmatrix} \tag{6}$$

where:

- n: represents the height or number of rows in the spatial grid.
- m: denotes the width or number of columns in the spatial grid.
- 1: denotes the number of channels.

3.3.5. Training the CNN with the Selected Features

One approach to integrate Random Forests (RF) with Convolutional Neural Networks (CNNs) involves utilizing the RF for feature selection. The RF can identify the most relevant features from the original data. Subsequently, these selected features can be used as input for a CNN model. This approach potentially improves the efficiency and effectiveness of the CNN. By focusing on the most informative features, the CNN might require less training data and reduce the risk of overfitting.

Let $\underline{X}$ of size $n \times m \times c$ represents the input tensor, where n and m are the spatial dimensions within the tensor structure, and c is the number of channels. Here, we used the matrix of selected-features from RF represented by $\underline{X}$ (after converting to a tensor) to train the Convolutional Neural Network (CNN). The mathematical expression of the convolutional layer is as follows:

$$Y_{l,j} = \sigma\left(\sum_{i \in S_j} \underline{X}_{l-1,i} \times_n w_{l,i,j} + b_{l,j}\right) \tag{7}$$

where:

- $Y_{l,j}$ is the output of the j^{th} neuron in the l^{th} layer.
- $\underline{X}_{l-1,i}$ refers to the input tensor of the i^{th} neuron in the l^{th} layer.
- $w_{l,i,j}$ represents the weights associated with the connections between the i^{th} neuron in the $(l-1)^{\text{th}}$ layer and the j^{th} neuron in the l^{th} layer. And n is the mode-n product.
- $b_{l,j}$ represents the bias associated with the j^{th} neuron in the l^{th} layer.
- S_j represents the set of indices i corresponding to the neurons in the $(l-1)^{\text{th}}$ layer that are connected to the j^{th} neuron in the l^{th} layer.

The expression $\underline{X}_{l-1,i} \times_n w_{l,i,j}$ is the *mode-n* product, which is an essential process in tensor algebra and its functions are used in multi-linear algebra and tensor factorization. Moreover, the mode-n product of a tensor and a matrix is executed along a specified mode or dimension of the tensor. The activation function ReLU is calculated as follows:

$$\sigma = \max\left(0, Y_{l,j}\right) \tag{8}$$

The mathematical formula of the Max Pooling layer is as follows:

$$MP_{l,j} = \max_{i \in R_j}\left(Y_{l,j}\right) \tag{9}$$

where:

- $M_{l,j}$ denotes the output value of the j^{th} neuron in the pooling layer.
- $\max_{i \in R_j}$ is the maximum operation over a specific region of indices R_j.

The mathematical formula of the fully connected layer will be:

$$Z_l = W_l \cdot MP_{l-1} + b_{l,j} \tag{10}$$

$$A_l = \sigma(Z_l) \tag{11}$$

where:

- Z_l is the output of the fully connected layer.
- A_l denotes the activation values of the neurons in the fully connected layer.
- MP_{l-1} is the output of the max pooling operation at layer $l-1$.
- W_l denotes the weight matrix associated with the connections between the previous layer (MP_{l-1}) and the current layer (Z_l)

Let us assume that the last fully connected layer is denoted as A_L, where L represents the total number of layers in the network.

The final output of the CNN will be:

$$Y_{cnn} = linear(Z_L) \tag{12}$$

where:

Z_L represents the input to the activation function of the last fully connected layer.

Y_{cnn} is the final output of the CNN.

In the illustrated CNN architecture (**Figure 3**), the initial layer corresponds to the input vector space, representing the input data. This architecture comprises two hidden layers, each utilizing convolutional and pooling operations to extract spatial features from the input data. Finally, the linear output layer transforms the features learned by the convolutional layers into a final prediction using activation functions.

3.3.6. The Hybrid Model

Hybrid machine learning models, integrating components from diverse model types or learning algorithms, have become widely embraced for tackling complicated challenges and enhancing overall effectiveness. They have emerged in response to the growing complexities of real-world challenges, aiming to enhance performance. With ongoing technological advancements, the field of hybrid machine learning models is witnessing further innovation and refinement. According to [37], a hybrid model is an approach that involves utilizing the probabilities generated by one machine learning model as input for another machine learning model, aiming to achieve better-optimized results based on both machine learning procedures, which are considered for the implementations.

In this part, we will combine the selected features X_{tr}, the predictions of RF ($\hat{Y}_{rf}^{D}(x)$) and CNN (Y_{cnn}) to create the hybrid model. These combined features serve as richer input data for Adaboost, potentially enhancing its ability to capture complex patterns and improve overall performance, as shown in **Figure 4**.

In order to increase the performance of our hybrid model, we will use the Adaboost ensemble algorithm. Adaboost, or Adaptive Boosting, builds a model and gives equal weights to all the data points. It then assigns higher weights to points that are wrongly classified. Adaboost improves the performance of the model by combining several weak learners' accuracies [38].

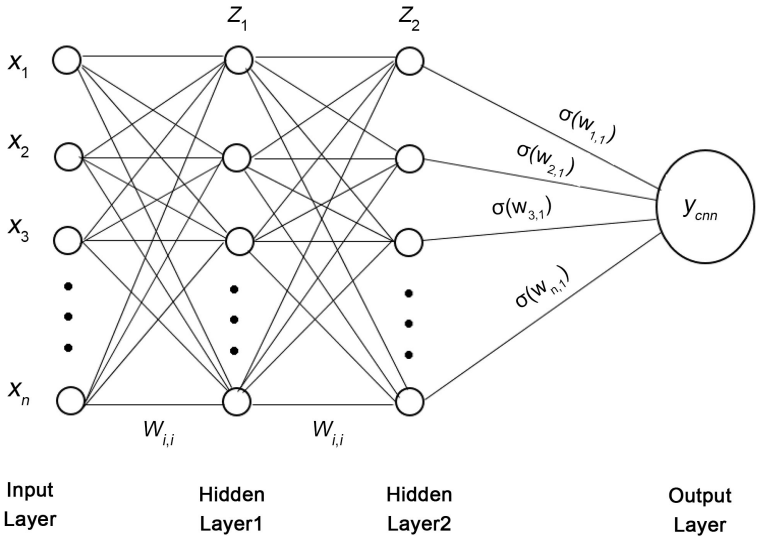

Figure 3. The CNN architecture.

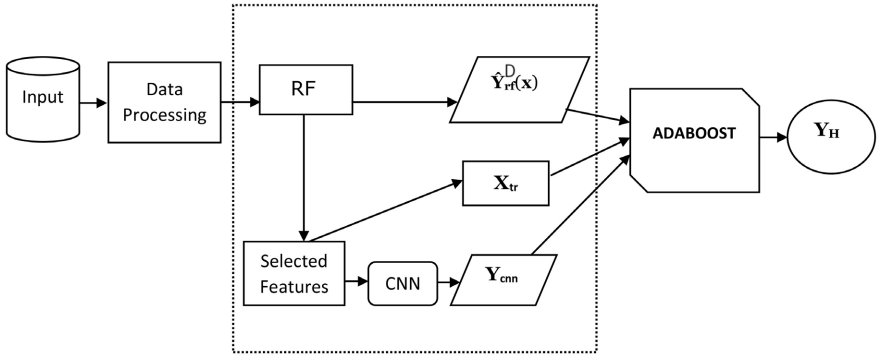

Figure 4. Flow diagram of the proposed hybrid ensemble model.

Let X_{tr} represents the set of selected features by the RF from a training dataset with n samples and m features, denoted as: $(x_1, y_1), (x_2, y_2), \cdots, (x_n, y_n)$, where x_i represents the feature values and y_i the corresponding outputs.

Let $\hat{Y}_{rf}^{D}(x)$ be the output of the Random Forest Regressor, and let Y_{cnn} be the output of the convolutional Neural network (CNN). Then, the combined features F can be represented as the concatenation of the selected features X_{tr}, the predictions of RF ($\hat{Y}_{rf}^{D}(x)$), and the predictions of CNN (Y_{cnn}). Mathematically, this can be written as:

$$F = \left[X_{tr} + \hat{Y}_{rf}^{D}(x) + Y_{cnn} \right] \tag{13}$$

Let us define by $\mathcal{S}$ the loss function:

$$\mathcal{S}(y_i, Y_H) \tag{14}$$

Given: y_i the ground true labels and Y_H the predictive model, in this case our hybrid model;

The goal is to have a minimal value for the loss function:

$$\mathcal{S}(y_i, Y_H) = |y_i - Y_H| \tag{15}$$

Initialize weights $w_i = \frac{1}{N}$, $i = 1, 2, \cdots, N$, where N represents the number of samples.

For $t = 1, 2, \cdots$ to T:

Fit a weak learner $l_t(x)$ to the combined features F.

T denotes the number of iterations or the total number of weak learners trained during the ensemble learning process.

$$l_t = H(F, F_t) \tag{16}$$

where:

- l_t represent a weak learner;
- H is the base learning algorithm;
- F is the combined features;
- F_t is a distribution of F or a subset of training samples.

Calculate the error er_t

$$er_t = \frac{\sum_{i=1}^{N} w_i^{(t)} \left| y_i - l_t(x_i) \right|}{\sum_{i=1}^{N} w_i} \tag{17}$$

where:

- er_t is the error of the t^{th} weak learner.
- N represents the total number of samples.
- w_i represents the weight associated with the i^{th} sample. These weights are updated at each iteration of the AdaBoost algorithm.
- y_i denotes the true label of the i^{th} sample.
- $l_t(x_i)$ is the prediction of the t^{th} weak learner for the i^{th} sample.

Set the performance of stump

$$\lambda_t = \frac{1}{2} \ln\left(\frac{1 - er_t}{er_t}\right) \tag{18}$$

where:

- λ_t represents the performance coefficient associated with the t^{th} weak learner.
- er_t is the error rate of the t^{th} weak learner, which is calculated in Equation 17.

Update the weights

$$w_i^{t+1} = w_i^t \cdot \exp\left(-\lambda_t \cdot \mathcal{S}\left(y_i - l_t(x)\right)\right) \tag{19}$$

where:

$\left(-\lambda_t \cdot \mathcal{S}\left(y_i - l_t(x)\right)\right)$ increases the weights for samples with larger errors and decreases for those with smaller errors.

Normalize w_i to sum to one

$$w_i^{t+1} = \frac{w_i^{t+1}}{\sum_{j=1}^{N} w_j^{(t+1)}} \tag{20}$$

where:

$w_i^{(t+1)}$ denotes the weight assigned to the sample i after an update has been applied in iteration $t+1$, N the number of samples.

$$l_t(x) = l_{t-1}(x) + \lambda_t \cdot l_t(x) \tag{21}$$

where:

- $l_{t-1}(x)$ is the ensemble model built up to the $(t-1)^{\text{th}}$ iteration.
- $l_t(x)$ is the updated ensemble model after incorporating the t^{th} weak learner's contribution.

The final ensemble prediction is computed by combining the predictions of all weak learners weighted by their corresponding coefficients λ_t, as shown in the formula (22).

$$Y_{\mathbf{H}} = \sum_{i=1}^{T} \lambda_t \cdot l_t(x) \tag{22}$$

where:

- $Y_{\mathbf{H}}$ is the prediction of the hybrid model.
- λ_t is the performance of the stump.
- $l_t(x)$ represents the weaker learner.

3.4. Cross-Validation

Cross validation is a strategy used in machine learning to estimate the performance of a model on unseen data. It requires splitting the provided data into multiple folds or subsets, using one of these folds as a validation set, and training the model on the lasting folds. Cross-validation has the computational advantage that it avoids fitting a model too closely to the particularities of a dataset (overfitting) [39]. There exist multiple types of cross-validation but in this study we are going to use the *k*-fold cross-validation. *K*-fold cross validation is a method used to assess predictive models. It involves splitting the dataset into *k* subsets or folds. The model is then trained and tested *k* times, with each fold serving as the validation set in turn. The performance metrics are collected from each fold and averaged to gauge how well the model generalizes [40]. Mathematically, this can be represented as:

$$\mathcal{E}(k) = \frac{1}{k} \sum \mathcal{S}\left(y_i, \hat{p}_{-k}(x_i)\right) \tag{23}$$

where:

$\mathcal{E}$ is the average loss across k folds, k represents the number of folds in the cross-validation process, $\mathcal{S}$ represents the loss function for the i^{th} fold, $\hat{p}_{-k}(x_i)$ denotes the predicted value by the model trained on all data points except those in the k^{th} fold, and y_i represents the true labels.

3.5. Spatial Auto-Correlation

Spatial autocorrelation refers to the correlation between data points that arises exclusively from their proximity in space. Positive spatial auto-correlation manifests when similar values cluster together, while negative spatial auto-correlation

manifests when dissimilar values are closer together. There is no spatial auto-correlation when the values of a variable are randomly distributed across space, with no observable pattern of similarity or dissimilarity between neighboring geographic units. In this paper, we utilized both Global and Local Moran's I statistics to investigate spatial auto-correlation in our dataset. Global Moran's I provide an overall assessment of spatial auto-correlation across the entire study area, indicating whether similar values cluster together or are dispersed, it ranges from −1 (showing perfect dispersion) to 1 (showing perfect clustering), with values close to zero suggesting no spatial auto-correlation [41]. Conversely, Local Moran's I allows for the identification of spatial clusters and outliers by evaluating auto-correlation at the local level.

3.6. Models Evaluation

Model evaluation is important to assess the efficacy of a model during initial research steps, and it also plays a role in model monitoring. The evaluation metrics used in this research study are the Mean absolute Error (MAE), Root Mean Squared Error (RMSE) and the coefficient of determination (R^2). These will be examined in detail below:

Mean Absolute Error

The mean absolute error serves as a statistical metric for measuring the average absolute differences between predicted and actual values, providing a simple way of assessing the accuracy of a predictive model. The MAE is presented in the same units as the data facilitating straightforward interpretation. A lower MAE indicates a closer alignment between the model's predictions and the actual values, signifying better predictive accuracy [42]. Mathematically, it can be calculated as follows:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}\left|y_i - \hat{y}_i\right| \tag{24}$$

where:

n denotes the total number of data points, y_i represents the observed (actual) value, $\hat{y}_i$ is the predicted value.

Root Mean Squared Error

The RMSE represents the normal distribution of prediction errors. These errors, also called residuals demonstrate the distance of the observations from the regression line, and RMSE works as measure of how those residuals are spread or dispersed [43]. Equation (25) describes the error function as presented below:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2} \tag{25}$$

where:

n denotes the total number of data points, y_i represents the actual value, $\hat{y}_i$ is the predicted value. The final loss is determined by computing the squared differences between them and subsequently summing these squared values.

When measuring the accuracy, the square root is taken over this summation.

R-squared

R-squared, denoted as (R^2), is a frequently used statistical measure. It quantifies the proportion of variability in the dependent variable (y) that can be attributed to the independent variable (x) in regression models. When R^2 values fall within the range of 0 to 1, it signifies that they span from 0% to 100% of the variation in the vertical axis, contingent on the values observed on the horizontal axis [44]. Equation (26) represents that measure:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2}{\sum_{i=1}^{n}\left(y_i - \bar{y}\right)^2} \tag{26}$$

where:

- R^2 is the coefficient of determination.
- $\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2$ represents the sum of squared residuals, which referred to the summation of the squared variances between the observed values (y_i) and the predicted values ($\hat{y}_i$).
- $\sum_{i=1}^{n}\left(y_i - \bar{y}\right)^2$ denotes the total sum of squares, which is the sum of squared variances between the observed values (y_i) and the mean of these observed values, symbolized as ($\bar{y}$).

4. Experimental Results

4.1. Feature Selection Results

The result obtained from feature importance analysis using the Random Forest algorithm provides valuable insights into the significance of different features in the dataset. **Figure 5** shows the result of the Feature Selection. We noticed that the score of the housing_median_age is far different from the first four features' scores so we considered these four features. In this case, it's important to observe that the "median_income" feature emerges as the most important one, which makes sense as areas where rich people live tend to have more expensive houses. Following "median_income", the "ocean_proximity" feature is ranked second in importance. This ranking implies that the price of a house depends on its geographical position relative to the ocean. The third-ranking feature is "longitude", suggesting that the location in terms of east-west position also plays a significant role in determining house prices. Finally, the fourth most important feature is "latitude", indicating that the north-south position of a house also influences its price.

4.2. Machine Learning Model Results

Before constructing the Hybrid model, we conducted thorough hyper parameter tuning for both the Random Forest (RF) and the Convolutional Neural Network (CNN) models. **Table 3** shows the best hyper-parameters for each model. For the RF model, we employed a randomized search approach due to the large search space to optimize parameters such as the number of trees, minimum

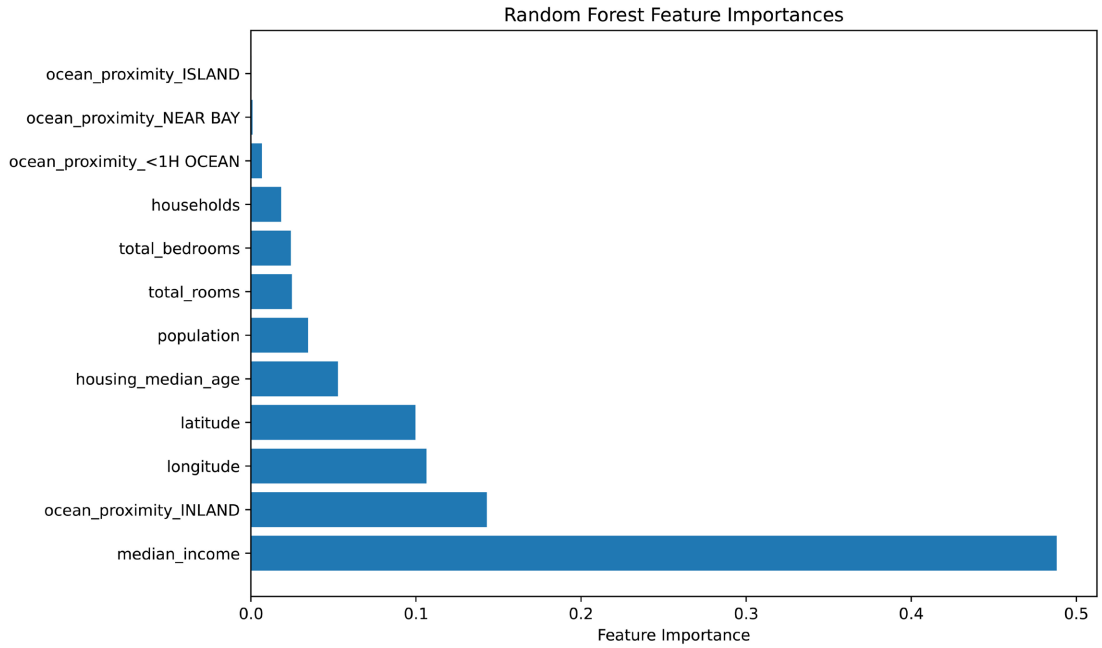

Figure 5. Feature Importance by RF.

Table 3. Best Hyper-parameters.

Model	Hyperparameters
Random Forest (RF)	Number of Estimators: 400 Minimum Samples Split: 5 Minimum Samples Leaf: 1 Maximum Features: "sqrt" Maximum Depth: 30 Bootstrap: True
Convolutional Neural Network (CNN)	Optimizer: Adam Learning Rate: 0.001 Epochs: 50 Dropout Rate: 0.5 Dense Units: 256 Dense Layers: 2 Convolutional Layers: 1 Convolutional Kernel Size: 5 Convolutional Filters: 128 Batch Size: 64 Activation Function: ReLU
Hybrid Model	Base Estimator Maximum Depth: 7 Learning Rate: 0.01 Number of Estimators: 200

samples split, minimum samples leaf, maximum features, maximum depth, and bootstrap. By enabling bootstrapping (True), random samples are drawn with replacement from the training set, fostering diversity and controlling overfitting. Overall, these hyperparameters are chosen to balance model complexity and performance, aiming to achieve a robust and well-generalized Random Forest model.

Similarly, for the CNN model, we utilized a randomized search over predefined parameter grid due to the large search space to find the optimal configuration of convolutional layers, pooling layers, learning rates, dropout rates, dense units, and other relevant parameters. The Adam optimizer is chosen for its adaptive learning rate and momentum properties, which can lead to faster convergence and better performance. As a whole, the chosen hyperparameters reflect a thoughtful approach to designing the CNN architecture, balancing between model complexity, training stability, and generalization performance.

Next, the hybrid ensemble model (Adaboost) was created using the selected features and the predictions from RF and CNN as additional features, then, since the search space is not large compared to RF and CNN, we employed a grid search to optimize parameters such as max_depth, learning_rate, and the number_of_estimators for the hybrid ensemble model. These parameter selections aim to find an equilibrium between model intricacy and its ability to accurately forecast outcomes, ensuring a well-balanced trade-off between complexity and performance.

Table 4 presents a comparison of the performance results among Random Forest (RF), Convolutional Neural network (CNN), and the proposed hybrid model, based on the RMSE, MAE, and R^2 scores.

The hybrid model's superior performance in this study derives from several factors. Firstly, upon comparing metrics like mean absolute error (MAE), root mean squared error (RMSE), and coefficient of determination (R^2) in Table 4, it's evident that the hybrid model outperforms both the random forest (RF) and convolutional neural network (CNN) models in all aspects. This implies that the hybrid model achieves greater accuracy and predictive capacity. Additionally, the assessment of global spatial autocorrelation in Table 6 using Moran's I on residuals shows that the hybrid model displays a lower Moran's I value compared to both RF and CNN models, indicating reduced spatial autocorrelation in prediction errors, and suggesting that the hybrid model effectively captures spatial dependencies in the data, leading to more accurate spatial predictions. Overall, integrating features and methodologies from both RF and CNN models enhances the hybrid approach's performance and spatial prediction capabilities.

Table 4. Results of different models on the testing set.

Models	MAE	RMSE	R^2
RF	0.25945	0.40136	0.83557
CNN	0.36000	0.50608	0.73858
Hybrid	0.18600	0.24112	0.90058

4.3. K-Fold Cross-Validation Results

To underscore the significance of our proposed model, we conducted a 5-fold cross-validation analysis using all models on the spatial dataset in use. Table 5 presents the RMSE performance metrics obtained from cross-validation process for each model, alongside their respective test errors. Upon examining the spatial data, it becomes clear that the hybrid model consistently achieves the lowest RMSE values across all folds and demonstrates the lowest test error, which represents the average RMSE across all folds within the model. These results strongly suggest that the hybrid model is the most accurate for the given task. Additionally, the RF model displays commendable performance, exhibiting lower RMSE values compared to the CNN model. Therefore, the CNN model shows moderately higher RMSE values, indicating relatively lower accuracy in predicting the house prices.

4.4. Spatial Auto-Correlation Results

In our spatial auto-correlation assessment, we employed the libpysal library in **Python**, utilizing its KNN function to compute the spatial weights based on the k-nearest neighbors approach. This analysis was conducted on a GeoDataFrame gdf_train comprising spatial observations, with k values ranging from 1 to 5 being evaluated. For each k value, we computed the spatial lag, representing the weighted average of neighboring values for a given variable. Subsequently, we assessed the spatial autocorrelation of residuals from the hybrid model by calculating the Pearson correlation coefficient, indicating strong spatial autocorrelation capture. Our analysis identified k = 5 as the optimal value, effectively representing spatial relationships within the dataset. Table 6 presents the outcomes of the Global Moran's I statistic for the residuals across various models.

The Global Moran's I value serves as an indicator of spatial autocorrelation, showing both the extent of spatial clustering or similarity and the presence of dissimilarities within the residuals. For the CNN model, a Moran's I value of 0.48 suggests a notably strong positive spatial autocorrelation pattern. Conversely, the RF model exhibits a lower Moran's I value of 0.12, indicating a relatively weaker spatial autocorrelation contrasted to the CNN model. Notably, the hybrid model displays the lowest Moran's I value of 0.10, suggesting minimal spatial autocorrelation among the models examined. Overall, the comparison of Moran's I values underscores a reduction in spatial autocorrelation with the residuals, highlighting the influence of model performance on capturing spatial autocorrelation.

Table 5. 5-fold cross validation results.

Models	RMSE scores across outer folds for various models					Test Error
	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	
RF	0.40653	0.42186	0.40352	0.42007	0.41772	0.40136
CNN	0.52449	0.53675	0.53037	0.50402	0.50808	0.50608
Hybrid	0.18889	0.18788	0.17808	0.18807	0.18857	0.18600

Table 6. Assessment of global spatial autocorrelation.

Models	Moran's I on residuals
RF	0.12
CNN	0.48
Hybrid	0.10

Additionally, the study examined Local Moran's I values to further understand spatial autocorrelation patterns at a more localized level. Positive values indicate clusters of high or low residuals surrounded by similar values, suggesting spatial homogeneity. Conversely, negative values indicate clusters of high residuals surrounded by low residuals (or vice versa), highlighting spatial discontinuities or areas of spatial heterogeneity within the dataset. The Local Moran map for the CNN model **Figure 6** reveals significant spatial heterogeneity in residual values across the study area. The map displays a diverse range of colors, each representing varying intensities of residuals. Blue and purple areas indicate localized clusters of low residuals, while brown and orange areas signify localized clusters with higher residuals. Additionally, the presence of yellow areas denotes regions with substantially higher residuals. Overall, the map underscores the complex and varied nature of residual values observed across the study area, emphasizing the spatial heterogeneity and variability inherent in the CNN model's predictions.

Similarly, **Figure 7** demonstrates significant spatial heterogeneity in residual values across the study area, showcasing a diverse color range indicating variability in intensities. Blue and purple areas suggest localized clusters of low residuals, while orange and yellow areas denote clusters with higher residuals. Additionally, brown areas indicate dispersed regions with near-zero residuals. This variability underscores the complex nature of residual values across the area, reflecting both spatial homogeneity and distinct patterns. Finally, The Local Moran map for the hybrid model **Figure 8** provides insights into the spatial distribution in residual values across the study area. The hybrid map demonstrates a wider range of colors and intensities, indicating a finer resolution in capturing subtle spatial variations compared to RF and CNN. Purple and navy blue areas highlight localized clusters of low residuals, while green and light green areas denote clusters with slightly higher values. The presence of yellow areas indicates regions with significantly higher residuals. This diverse range of colors underscores the complex and varied nature of residual values observed across the study area, emphasizing the overall spatial heterogeneity inherent in the hybrid model's predictions. Among the three models, the hybrid model appears to capture spatial heterogeneity more effectively. This inference is drawn from the distribution of Local Moran's I values, where the hybrid model demonstrates a balanced mix of positive and negative values, indicating the presence of both spatial clusters of similar and dissimilar values. However, the hybrid model offers a com-

prehensive approach to understanding spatial heterogeneity. This integration allows the hybrid model to effectively capture diverse spatial patterns and dependencies, making it a superior model in characterizing the complex spatial structure of the dataset.

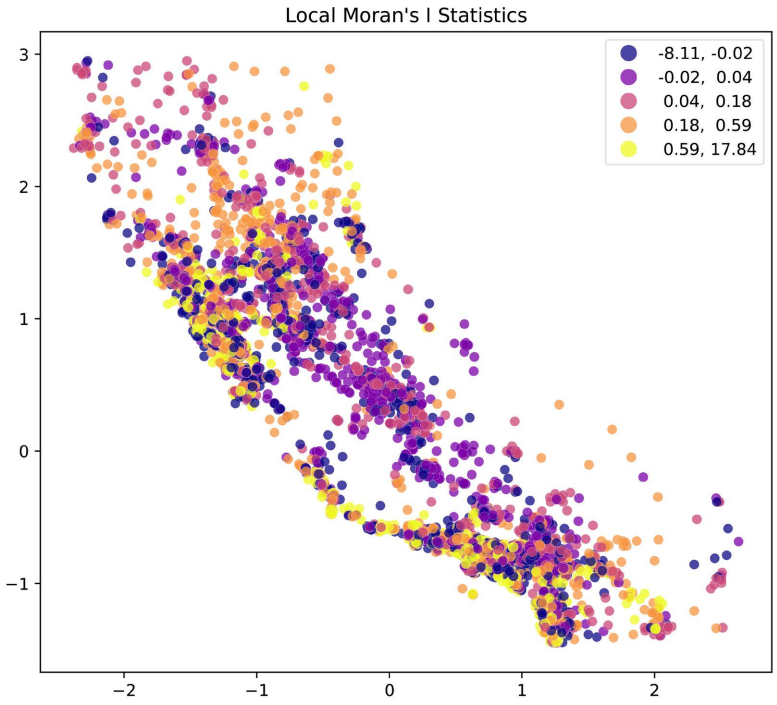

Figure 6. CNN model.

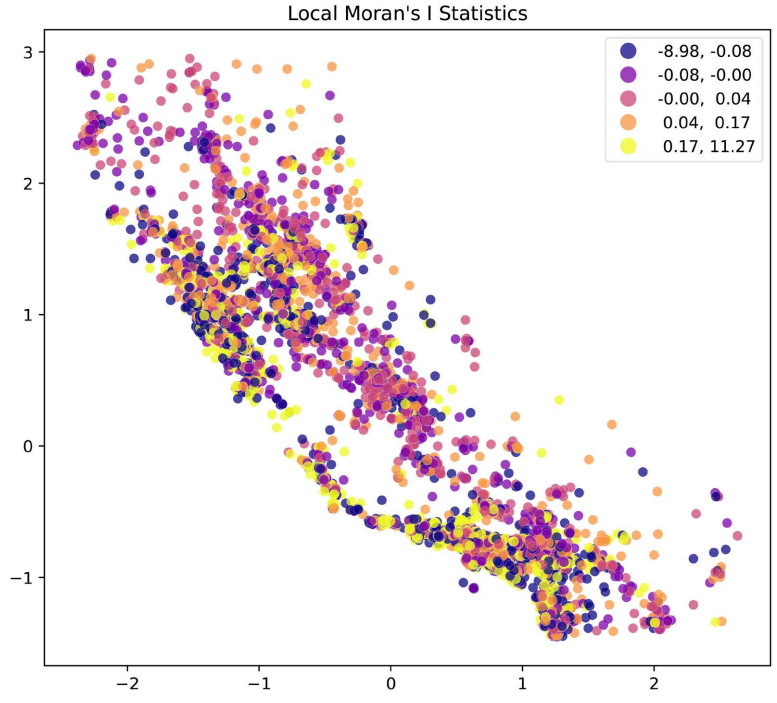

Figure 7. RF model.

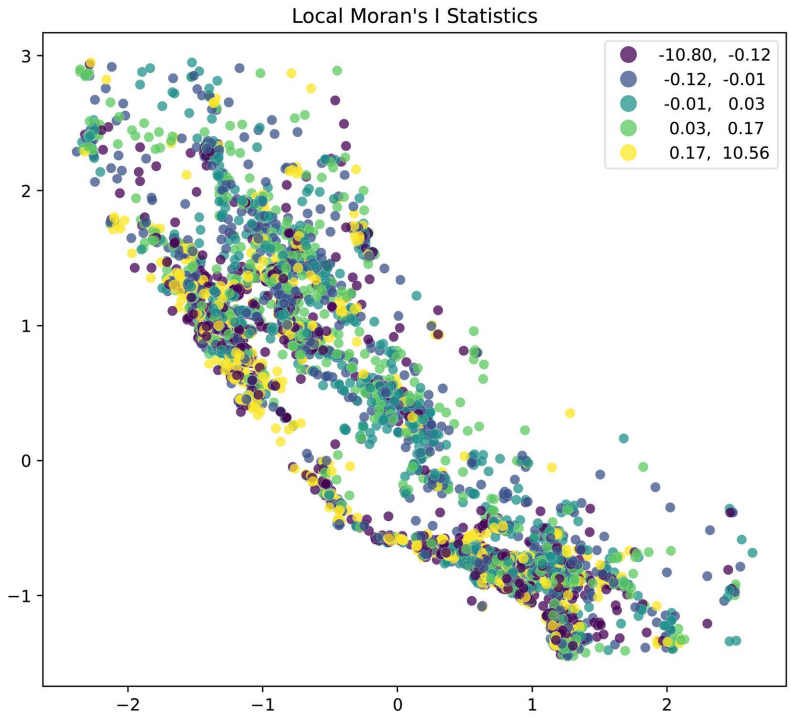

Figure 8. Hybrid model.

The analysis indicates that the models have effectively captured the spatial characteristics, as reflected in the robust performance assessment based on RMSE for the hybrid model and the fair performance for the remaining two models. These findings are consistent with our initial hypotheses, illustrating the capability of a particular ML model to handle spatial heterogeneity without the need for explicit spatial features in the learning phase. Furthermore, it underscores the model's aptitude for capturing spatial relationships and enhancing predictive accuracy, thus enriching our comprehension of spatial heterogeneity.

5. Discussion

Our study presents a novel approach to spatial heterogeneity detection using machine learning (ML) techniques without the need for explicit spatial features during the learning phase. By combining Random Forest (RF) and Convolutional Neural Network (CNN) models into a hybrid model, we aimed to achieve superior performance in capturing spatial patterns and dependencies. The performance evaluation as depicted in **Table 4** reveals that the hybrid model outperformed both RF and CNN models in terms of MAE, RMSE and R^2 scores. Notably, the hybrid model achieved a remarkable R^2 score of 0.90058, indicating its ability to capture 0.90% of the variance between predicted and actual values. These results underscore the robustness and efficacy of our hybrid approach in accurately modeling housing prices. This study aligns with [45] [46]. Cross-validation results (**Table 5**) further validate the superiority of the hybrid model, as it consistently demonstrated the lowest RMSE values across all folds. This

consistency highlights the hybrid model's accuracy and reliability in capturing spatial dependencies within the dataset. The analysis of spatial autocorrelation patterns using Global Moran's I statistics and Local Moran's I values revealed insightful findings. While the RF model exhibited a lower Global Moran's I value compare to CNN, the hybrid model demonstrated the lowest Global Moran's I value, indicating minimal spatial autocorrelation among the models examined. Common spatial autocorrelation analysis has been done in [47] but the hybrid approach and spatial autocorrelation based on models' residuals are more emphasis in our study. Combining predictions from multiple models, such as RF and CNN, using ensemble techniques to leverage the strengths of different models can help mitigate the effects of spatial heterogeneity, thus reducing prediction bias. Additionally, evaluating the spatial autocorrelation of model residuals using techniques like Global Moran's I or local Moran's I to identify any remaining spatial patterns or dependencies in model predictions that may require further attention.

In summary, our study demonstrated the effectiveness of our hybrid model in capturing spatial heterogeneity without the need for explicit spatial features. By integrating RF and CNN strengths, the hybrid model offers a comprehensive approach to understanding spatial patterns and dependencies. These findings contribute to advancing spatial modeling methodologies and hold significant implications for various applications, including urban planning and environmental management. Future research could explore additional ML algorithms and spatial modeling techniques to further enhance predictive accuracy and uncover deeper insights into spatial phenomena. Additionally, further investigation could consider applying our method with other ensemble techniques and spatial autocorrelation analyses to comprehensively explore the issue of spatial heterogeneity and its implications.

6. Conclusion

The current study presents a fresh approach to identifying spatial heterogeneity using machine learning (ML) methodologies, notably without requiring explicit spatial features during the learning phase. The proposed hybrid model combines two competitive models, Random Forest (RF) and Convolutional Neural Network (CNN), to achieve high accuracy and effectiveness. Both RF and CNN individually exhibit robust performance in discerning spatial relationships. However, their combination within the hybrid model resulted in a notable enhancement, marking a substantial 0.90% increase in accuracy. This improvement is credited to the application of boosting techniques, particularly, the Adaboost algorithm, which effectively rectifies errors inherent in each individual model. Noteworthy is that both individual models and the hybrid model proficiently captured significant spatial details, encompassing spatial dependencies as quantified by Global and Local Moran indices. Although showing lower R^2 values relative to the hybrid model, the separate models successfully encapsulated a substantial

portion of spatial information. In contrast, the hybrid model characterized by its high R^2 value, emerged as the superior model in capturing the spatial heterogeneity inherent within the dataset.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Goodchild, M.F. (2013) The Quality of Big (Geo) Data. *Dialogues in Human Geography*, **3**, 280-284. https://doi.org/10.1177/2043820613513392

[2] Gaspard, G., Kim, D. and Chun, Y. (2019) Residual Spatial Autocorrelation in Macroecological and Biogeographical Modeling: A Review. *Journal of Ecology and Environment*, **43**, Article No. 19. https://doi.org/10.1186/s41610-019-0118-3

[3] Shekhar, S., Zhang, P. and Huang, Y. (2010) Spatial Data Mining. In: Maimon, O. and Rokach, L., Eds., *Data Mining and Knowledge Discovery Handbook*, Springer, Berlin, 837-854. https://doi.org/10.1007/978-0-387-09823-4_43

[4] Dutilleul, P. and Legendre, P. (1993) Spatial Heterogeneity against Heteroscedasticity: An Ecological Paradigm versus a Statistical Concept. *Oikos*, **66**, 152-171. https://doi.org/10.2307/3545210

[5] Brenning, A. (2005) Spatial Prediction Models for Landslide Hazards: Review, Comparison and Evaluation. *Natural Hazards and Earth System Sciences*, **5**, 853-862. https://doi.org/10.5194/nhess-5-853-2005

[6] Aguilar, R., Zurita-Milla, R., Izquierdo-Verdiguier, E. and De By, R.A. (2018) A Cloud-Based Multi-Temporal Ensemble Classifier to Map Smallholder Farming Systems. *Remote Sensing*, **10**, Article No. 729. https://doi.org/10.3390/rs10050729

[7] Pradhan, A.M.S. and Kim, Y.-T. (2020) Rainfall-Induced Shallow Landslide Susceptibility Mapping at Two Adjacent Catchments Using Advanced Machine Learning Algorithms. *ISPRS International Journal of Geo-Information*, **9**, Article No. 569. https://doi.org/10.3390/ijgi9100569

[8] Zurita-Milla, R., Goncalves, R., Izquierdo-Verdiguier, E. and Ostermann, F.O. (2019) Exploring Spring Onset at Continental Scales: Mapping Phenoregions and Correlating Temperature and Satellite-Based Phenometrics. *IEEE Transactions on Big Data*, **6**, 583-593. https://doi.org/10.1109/TBDATA.2019.2926292

[9] Reichstein, M., Camps-Valls, G., Stevens, B., Jung, M., Denzler, J., Carvalhais, N. and Prabhat, F. (2019) Deep Learning and Process Understanding for Data-Driven Earth System Science. *Nature*, **566**, 195-204. https://doi.org/10.1038/s41586-019-0912-1

[10] Shekhar, S., Jiang, Z., Ali, R.Y., Eftelioglu, E., Tang, X., Gunturi, V.M.V. and Zhou, X. (2015) Spatiotemporal Data Mining: A Computational Perspective. *ISPRS International Journal of Geo-Information*, **4**, 2306-2338. https://doi.org/10.3390/ijgi4042306

[11] Nwaila, G.T., Zhang, S.E., Bourdeau, J.E., Frimmel, H.E. and Ghorbani, Y. (2024) Spatial Interpolation Using Machine Learning: from Patterns and Regularities to Block Models. *Natural Resources Research*, **33**, 129-161. https://doi.org/10.1007/s11053-023-10280-7

[12] Wang, Z., Shi, W.J., Zhou, W., Li, X.Y. and Yue, T.X. (2020) Comparison of Addi-

tive and Isometric Log-Ratio Transformations Combined with Machine Learning and Regression Kriging Models for Mapping Soil Particle Size Fractions. *Geoderma*, **365**, Article ID: 114214. https://doi.org/10.1016/j.geoderma.2020.114214

[13] Pereira, G.W., *et al.* (2022) Smart-Map: An Open-Source QGIS Plugin for Digital Mapping Using Machine Learning Techniques and Ordinary Kriging. *Agronomy*, **12**, Article No. 1350. https://doi.org/10.3390/agronomy12061350

[14] Hengl, T., Nussbaum, M., Wright, M.N., Heuvelink, G.B.M. and Graler, B. (2018) Random Forest as a Generic Framework for Predictive Modeling of Spatial and Spatio-Temporal Variables. *PeerJ*, **6**, e5518. https://doi.org/10.7717/peerj.5518

[15] Behrens, T., Rossel, R.A.V., Kerry, R, MacMillan, R., Schmidt, K., Lee, J., Scholten, T. and Zhu, A.-X. (2019) The Relevant Range of Scales for Multi-Scale Contextual Spatial Modelling. *Scientific Reports*, **9**, Article No. 14800. https://doi.org/10.1038/s41598-019-51395-3

[16] Georganos, S., Grippa, T., Gadiaga, A.N., Linard, C., Lennert, M., Vanhuysse, S., Mboga, N., Wolff, E. and Kalogirou, S. (2021) Geographical Random Forests: A Spatial Extension of the Random Forest Algorithm to Address Spatial Heterogeneity in Remote Sensing and Population Modelling. *Geocarto International*, **36**, 121-136. https://doi.org/10.1080/10106049.2019.1595177

[17] Meyer, H., Reudenbach, C., Wollauer, S. and Nauss, T. (2019) Importance of Spatial Predictor Variable Selection in Machine Learning Applications-Moving from Data Reproduction to Spatial Prediction. *Ecological Modelling*, **411**, Article ID: 108815. https://doi.org/10.1016/j.ecolmodel.2019.108815

[18] Behrens, T. and Rossel, R.A.V. (2020) On the Interpretability of Predictors in Spatial Data Science: The Information Horizon. *Scientific Reports*, **10**, Article No. 16737. https://doi.org/10.1038/s41598-020-73773-y

[19] Chen, L., Ren, C.Y., Li, L., Wang, Y.Q., Zhang, B., Wang, Z.M. and Li, L.F. (2019) A Comparative Assessment of Geostatistical, Machine Learning, and Hybrid Approaches for Mapping Topsoil Organic Carbon Content. *ISPRS International Journal of Geo-Information*, **8**, Article No. 174. https://doi.org/10.3390/ijgi8040174

[20] Behrens, T., Schmidt, K., Rossel, R.A.V., Gries, P., Scholten, T. and MacMillan, R.A. (2018) Spatial Modelling with Euclidean Distance Fields and Machine Learning. *European Journal of Soil Science*, **69**, 757-770. https://doi.org/10.1111/ejss.12687

[21] Quinones, S., Goyal, A. and Ahmed, Z.U. (2021) Geographically Weighted Machine Learning Model for Untangling Spatial Heterogeneity of Type 2 Diabetes Mellitus (T2D) Prevalence in the USA. *Scientific Reports*, **11**, Article No. 6955. https://doi.org/10.1038/s41598-021-85381-5

[22] Liu, X.J., Kounadi, O. and Zurita-Milla, R. (2022) Incorporating Spatial Autocorrelation in Machine Learning Models Using Spatial Lag and Eigenvector Spatial Filtering Features. *ISPRS International Journal of Geo-Information*, **11**, Article No. 242. https://doi.org/10.3390/ijgi11040242

[23] Khaki, S., Wang, L.Z. and Archontoulis, S.V. (2020) A CNN-RNN Framework for Crop Yield Prediction. *Frontiers in Plant Science*, **10**, Article ID: 492736. https://doi.org/10.3389/fpls.2019.01750

[24] Yu, W.T., Li, J., Liu, Q.H., Zhao, J., Dong, Y.D., Wang, C., Lin, S.R., Zhu, X.R. and Zhang, H. (2021) Spatial-Temporal Prediction of Vegetation Index with Deep Recurrent Neural Networks. *IEEE Geoscience and Remote Sensing Letters*, **19**, 1-5. https://doi.org/10.1109/LGRS.2021.3064814

[25] Xu, L., Cai, R.N., Yu, H.C., Du, W.Y., Chen, Z.Q. and Chen, N.C. (2024) Monthly NDVI Prediction Using Spatial Autocorrelation and Nonlocal Attention Networks.

IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, **17**, 3425-3437. https://doi.org/10.1109/JSTARS.2024.3350053

[26] Deng, M., Yang, W.T. and Liu, Q.L. (2017) Geographically Weighted Extreme Learning Machine: A Method for Space-Time Prediction. *Geographical Analysis*, **49**, 433-450. https://doi.org/10.1111/gean.12127

[27] Deng, M., Yang, W.T., Liu, Q.L., Jin, R., Xu, F. and Zhang, Y.F. (2018) Heterogeneous Space-Time Artificial Neural Networks for Space-Time Series Prediction. *Transactions in GIS*, **22**, 183-201. https://doi.org/10.1111/tgis.12302

[28] Wang, Y.M., Feng, L.W., Li, S.J., Ren, F. and Du, Q.Y. (2020) A Hybrid Model Considering Spatial Heterogeneity for Landslide Susceptibility Mapping in Zhejiang Province, China. *Catena*, **188**, Article ID: 104425. https://doi.org/10.1016/j.catena.2019.104425

[29] Almulihi, A., Saleh, H., Hussien, A.M., Mostafa, S., El-Sappagh, S., Alnowaiser, K., *et al.* (2022) Ensemble Learning Based on Hybrid Deep Learning Model for Heart Disease Early Prediction. *Diagnostics*, **12**, Article No. 3215. https://doi.org/10.3390/diagnostics12123215

[30] Zeng, H.R., Zhang, B. and Wang, H.J. (2023) A Hybrid Modeling Approach Considering Spatial Heterogeneity and Nonlinearity to Discover the Transition Rules of Urban Cellular Automata Models. *Environment and Planning B*: *Urban Analytics and City Science*, **50**, 1898-1915. https://doi.org/10.1177/23998083221149018

[31] Zhao, Z.X., Wu, J.R., Cai, F.J., Zhang, S.T. and Wang, Y.-G. (2023) A Hybrid Deep Learning Framework for Air Quality Prediction with Spatial Autocorrelation during the COVID-19 Pandemic. *Scientific Reports*, **13**, Article No. 1015. https://doi.org/10.1038/s41598-023-28287-8

[32] Rodriguez-Galiano, V., Sanchez-Castillo, M., Chica-Olmo, M. and Chica-Rivas, M. (2015) Machine Learning Predictive Models for Mineral Prospectivity: An Evaluation of Neural Networks, Random Forest, Regression Trees and Support Vector Machines. *Ore Geology Reviews*, **71**, 804-818. https://doi.org/10.1016/j.oregeorev.2015.01.001

[33] Li, J., Heap, A.D., Potter, A. and Daniell, J.J. (2011) Application of Machine Learning Methods to Spatial Interpolation of Environmental Variables. *Environmental Modelling & Software*, **26**, 1647-1659. https://doi.org/10.1016/j.envsoft.2011.07.004

[34] Lee, H., Kim, J., Jung, S., Kim, M. and Kim, S. (2019) Case Dependent Feature Selection Using Mean Decrease Accuracy for Convective Storm Identification. 2019 *IEEE International Conference on Fuzzy Theory and Its Applications* (*IFUZZY*), New Taipei, 7-10 November 2019, 1-4.

[35] Zhu, Y.T., Brettin, T., Xia, F.F., Partin, A., Shukla, M., Yoo, H., Evrard, Y.A., Doroshow, J.H. and Stevens, R. (2021) Converting Tabular Data into Images for Deep Learning with Convolutional Neural Networks. *Scientific Reports*, **11**, Article No. 11325. https://doi.org/10.1038/s41598-021-90923-y

[36] Liu, X., Wang, X.G. and Matwin, S. (2018) Improving the Interpretability of Deep Neural Networks with Knowledge Distillation. 2018 *IEEE International Conference on Data Mining Workshops* (*ICDMW*), Singapore, 17-20 November 2018, 905-912. https://doi.org/10.1109/ICDMW.2018.00132

[37] Kavitha, M., Gnaneswar, G., Dinesh, R., Rohith Sai, Y. and Sai Suraj, R. (2021) Heart Disease Prediction Using Hybrid Machine Learning Model. 2021 *IEEE 6th International Conference on Inventive Computation Technologies* (*ICICT*), Coimbatore, 20-22 January 2021, 1329-1333. https://doi.org/10.1109/ICICT50816.2021.9358597

[38] Taufiqurrahman, A., Putrada, A.G. and Dawani, F. (2020) Decision Tree Regression with Adaboost Ensemble Learning for Water Temperature Forecasting in Aquaponic Ecosystem. 2020 *IEEE 6th International Conference on Interactive Digital Media* (*ICIDM*), 14-15 December 2020, 1-5. https://doi.org/10.1109/ICIDM51048.2020.9339669

[39] Hastie, T., Tibshirani, R., Friedman, J., Hastie, T., Tibshirani, R. and Friedman, J. (2009) Overview of Supervised Learning. In: Hastie, T., Tibshirani, R. and Friedman, J., Eds., *The Elements of Statistical Learning*: *Data Mining*, *Inference*, *and Prediction*, Springer, Berlin, 9-41.

[40] Nti, I.K., Nyarko-Boateng, O., Aning, J., *et al.* (2021) Performance of Machine Learning Algorithms with Different K Values in K-Fold Cross-Validation. *International Journal of Information Technology and Computer Science*, **13**, 61-71. https://doi.org/10.5815/ijitcs.2021.06.05

[41] Chen, Y.G. (2013) New Approaches for Calculating Moran's Index of Spatial Autocorrelation. *PLOS ONE*, **8**, e68336. https://doi.org/10.1371/journal.pone.0068336

[42] Nguyen, K.T., Nguyen, Q.D., Le, T.A., Shin, J. and Lee, K. (2020) Analyzing the Compressive Strength of Green Fly Ash Based Geopolymer Concrete Using Experiment and Machine Learning Approaches. *Construction and Building Materials*, **247**, Article ID: 118581. https://doi.org/10.1016/j.conbuildmat.2020.118581

[43] Kobayashi, K. and Us Salam, M. (2000) Comparing Simulated and Measured Values Using Mean Squared Deviation and Its Components. *Agronomy Journal*, **92**, 345-352. https://doi.org/10.2134/agronj2000.922345x

[44] Andreas, A., Mavromoustakis, C.X., Mastorakis, G. Mumtaz, S., Batalla, J.M. and Pallis, E. (2020) Modified Machine Learning Technique for Curve Fitting on Regression Models for COVID-19 Projections. 2020 *IEEE 25th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks* (*CAMAD*), 14-16 September 2020, 1-6. https://doi.org/10.1109/CAMAD50429.2020.9209264

[45] Zhang, B.Z., Duan, M., Sun, Y.F., Lyu, Y.T., Hou, Y.L. and Tan, T. (2023) Air Quality Index Prediction in Six Major Chinese Urban Agglomerations: A Comparative Study of Single Machine Learning Model, Ensemble Model, and Hybrid Model. *Atmosphere*, **14**, Article No. 1478. https://doi.org/10.3390/atmos14101478

[46] Barry, M.H., Nderu, L. and Gichuhi, A.W. (2023) A Hybrid Spatial Dependence Model Based on Radial Basis Function Neural Networks (RBFNN) and Random Forest (RF). *Journal of Data Analysis and Information Processing*, **11**, 293-309. https://doi.org/10.4236/jdaip.2023.113015

[47] Sun, Y.M., Ao, Z.Q., Jia, W.W., Xu, K., *et al.* (2021) A Geographically Weighted Deep Neural Network Model for Research on the Spatial Distribution of the Down Dead Wood Volume in Liangshui National Nature Reserve (China). *IForest-Biogeosciences and Forestry*, **14**, 353-361. https://doi.org/10.3832/ifor3705-014