Research Paper **Open Access**

Fatemeh Mostofi*, Vedat Toğan, Hasan Basri Başağa

Real-estate price prediction with deep neural network and principal component analysis

DOI 10.2478/otmcj-2022-0016
Received: April 18, 2022; Accepted: November 10, 2022

Abstract: Despite the wide application of deep neural networks (DNN) models, their application over small-sized real-estate price prediction is limited due to the reduced prediction accuracy and the high-dimensionality of the dataset. This study motivates small-sized real-estate agencies to take DNN-driven decisions using the available local dataset. To improve the high-dimensionality of real-estate price datasets and thus enhance the price-prediction accuracy of a DNN model, this paper adopts principal component analysis (PCA). The PCA benefits in improving the prediction accuracy of a DNN model are threefold: dimensionality reduction, dataset transformation and localisation of influential price features. The results indicate that, through the PCA-DNN model, the transformed dataset achieves higher accuracy (90%–95%) and better generalisation ability compared with other benchmark price predictors. The spatial and building age proved to have the most impact in determining the overall real-estate price. The application of PCA not only reduces the high-dimensionality of the dataset but also enhances the quality of the encoded feature attributes. The model is beneficial in real-estate and construction applications, where the absence of medium and big datasets decreases the price-prediction accuracy.

Keywords: principal component analysis, deep neural network, high-dimensional dataset, real-estate price prediction, stepwise regression

***Corresponding author: Fatemeh Mostofi,** Department of Civil Engineering, Karadeniz Technical University, P.O. Box 61080, Trabzon, Türkiye,
E-mail: fatemee.mostofi@gmail.com; 393989@ogr.ktu.edu.tr
Vedat Toğan and Hasan Basri Başağa, Department of Civil Engineering, Karadeniz Technical University, P.O. Box 61080, Trabzon, Türkiye

1 Introduction

House price prediction based on existing real-estate data serves as fundamentals for property appraisal and real-estate market evaluation. The final sale price of a property is an important profitability determinant within an investment portfolio. Predicting the real-estate value allows individuals, companies and governments to effectively devise their financial plans. During the project feasibility stage, gauging the construction cost with the predicted sale price of a building directly influences the built/no built decision, which in turn determines the existence of a construction project. At the national level, the expected real-estate values have an impact on the associated contract, easements acquisition and pricing policies. An early estimate of the real-estate price value facilitates decision-making about its construction, purchase and/or regulation for real-estate owners, construction professionals and governments. Despite the importance of the topic for construction professionals and its stakeholders, it has not received adequate attention from the construction management field and it is mainly discussed by literature on real-estate price prediction within the real estate, finance, economy and business studies (Rafiei and Adeli 2016). While the real-estate price models within the literature have not been critically investigated (Li and Shi 2011), little attention has been paid to the price-prediction models that can be utilised over high-dimensional datasets: data with several feature columns for explaining a limited number of records.

The real-estate price is influenced by a complex range of factors and thus a price predictor needs to incorporate this complexity in a computationally efficient manner (Khalafallah 2008; Park and Kwon Bae 2015; Chen et al. 2017; Hu et al. 2019; Ho et al. 2021). Thus, the existing literature implemented a variety of statistical and pattern recognition approaches over different real-estate pricing data with various features to simulate the influential factors that affect the final real-estate price value (Rafiei and Adeli 2016). Over the past decades, the machine learning (ML) approaches enhanced decision-making within different

applications such as predicting identifying safe worker behaviour (Patel and Jha 2015), forecasting the variations in monthly material prices (Shiha et al. 2020) and predicting the house price values (Jiang and Shen 2019; Wang et al. 2019). The success of ML approaches within different fields encouraged the development of house price models that automatically extract the price information from different real-world datasets. To this end, artificial neural network (ANN), a widely used ML algorithm, is applied over a wide range of applications such as classification and language translation. However, a shortcoming of ANN is its difficulty in training the high-dimensional dataset (Awad and Khanna 2015).

A deep neural network (DNN) is a popular and strong extension of ANN that learns through a hierarchical procedure (Awad and Khanna 2015), and its performance is highly dependent on the quality and volume of the dataset. Even though the abundance of data improves the ML and subsequently DNN performance, still there are real-world cases where the dataset size is relatively small. Within this context, real-estate price valuation for developing cities restricts the study space to boundaries of that specific location and therefore limits the possible number of price-data records. These limited real-estate price records are often described within a high-dimensional feature space. Despite its wide application for price prediction, DNN adoption in real-estate applications is limited due to the mostly high-dimensionality and inaccessibility of big data. The high-dimensionality complicates the DNN training and decreases its price-prediction accuracy. In this respect, principal component analysis (PCA) is an unsupervised ML method that is used to reduce high-dimensionality within the dataset. The concept of PCA was invented by Pearson (1901) and further developed by Hotelling (1933). PCA projects the high-dimensional dataset into a lower subspace, where principal components are explained through a set of uncorrelated features (Ayesha et al. 2020). Accordingly, PCA extracts key features while retaining essential variance in the original feature attributes. This enables the identification and ranking of the influential features that improve the prediction accuracy of ML.

On the other hand, auto-encoders are projecting the dataset into the latent space for a better understanding of the dataset by the model. In this regard, PCA can also function similar to an auto-encoder for projecting the dataset into the lower-dimensional space (Reddy et al. 2020) and thence improves the DNN's understanding of a given high-dimensional dataset. Moreover, from applicational perspectives, both DNN and PCA are simple-to-implement algorithms, but despite this simplicity, their application over categorical construction and real-estate price data is not addressed by existing literature. This study demonstrates that the implementation of PCA after encoding enhances the model prediction ability. In other words, the PCA transformation of the high-dimensional real-estate dataset enhances the price-prediction accuracy of the DNN model. This study integrates PCA with DNN (PCA-DNN) to improve the prediction accuracy of real-estate sale prices. The PCA transforms the dataset into a new set of datasets with different feature columns, in which most of the variance is concentrated at the primary principal components.

This study's motivation for adopting PCA within a hybrid PCA-DNN model for real-estate price prediction is threefold: dimensionality reduction, feature selection and data transformation before processing in the DNN model. This study shows the efficiency of the PCA-DNN model over a hard-to-fit dataset, that is small-sized, high-dimensional and comprised of several categorical attributes. To this end, 1,244 real-world house price records were collected from the central area of Trabzon, Türkiye. The price records are described through categorical and numerical features, representing a general state of a data structure that can be found in a typical real-estate website in Türkiye.

To validate the developed model, its performance is benchmarked against a conventional DNN, and a stepwise regression analysis (SRA) combined with DNN (SRA-DNN) models. In addition, the number of trainable parameters associated with the number of layers and neurons is customised for each of the three models. Besides, these models were iterated for different loss optimisation and activation functions. Thus, the optimum network architecture (for each DNN, SRA-DNN and PCA-DNN models) was configured to enable comparison and evaluation amongst these models. Hence, the price-prediction performance of PCA-DNN model is compared with the benchmark models, using mean square error (MSE) (Pal 2017), mean absolute error (MAE) (Pal 2017) and mean absolute percentage error (MAPE) (Kim and Kim 2016) metrics. As a result, the PCA-DNN model developed in this study is useful for house price-prediction applications, where each real-estate unit price is described by a high-dimensional dataset.

2 Research background

The past research established the complexity of predicting the real-estate price value due to its influence from a wide range of influential determinates, such as inflation

rate (Stukhart 1982; Poterba 1984; Qiao and Guo 2014), policy adjustment (Zheng and Yan 2017), environmental events (Yue et al. 2020), accessibility (Zhang et al. 2022) and urban (Zhang et al. 2022) and public transportation services (Wen et al. 2022). To date, several studies have explored artificial intelligence (AI) (Rafiei and Adeli 2016; Cao et al. 2018; Gondia et al. 2020) in a variety of construction and real-estate predictions applications. As a branch of AI, ML approaches were utilised over the real-estate dataset to evaluate the most influential house price determinates (Zhai et al. 2018; Zhang et al. 2020; Luo et al. 2021). In a recent study, the empirical house price data from China are operated with a Hedonic price model for the identification of influential price determinants (Luo et al. 2021). Furthermore, the real-estate price prediction was exercised by adopting different ML approaches such as multiple linear regression (Zhang 2021), ANNs (Khalafallah 2008; Claesen and De Moor 2015), multi-layer perceptron NN (MLP-NN) (Hu et al. 2019), Naive Bayes (NB) (Park and Kwon Bae 2015), decision trees (DTs) (Park and Kwon Bae 2015; Peng and Wang 2022), K-nearest neighbour (KNN) (Hu et al. 2019) and support vector machine (SVM) (Chen et al. 2017; Hu et al. 2019; Ho et al. 2021). The research has found that grouping the benefit of single ML approaches within a single predictor (ensemble model) facilitates the creation of a more robust predictor compared with the contributing single ML techniques. In this regard, real-estate price-prediction was experimented with mainly tree-based ensemble approaches such as random forest (RF) (Sanjar et al. 2020; Ho et al. 2021; Chen et al. 2022; Peng and Wang 2022), RF with extra trees (Hu et al. 2019), AdaBoosting (AB) (Park and Kwon Bae 2015), gradient boosting (GB) (Hu et al. 2019; Ho et al. 2021; Chen et al. 2022), and XGBoost (Abdul-Rahman et al. 2021).

With the increase in computational power, the ANN concept was extended to structure a stronger predictor known as DNN. Compared to ANN, DNN increases the number of hidden layers while using a backpropagation mechanism for hierarchically learning from the precedent layers (Awad and Khanna 2015). The adoption of DNN models for house price prediction has recently attracted researchers' attention (Seya and Shiroi 2021), and applications of various DNN-based predictors were explored in the literature, such as convolutional neural network (CNN) (Piao et al. 2019) and long short-term memory (LSTM) (Kim et al. 2020). Seya and Shiroi (2021) showed that the prediction accuracy of DNN only improved when data size increased from order 4 ($n = 10^4$) to order 5 ($n = 10^5$), reaching a 90% MAPE value. However, their study is limited to medium-sized and large-sized datasets, and they have not investigated DNN application in the small-sized real-estate dataset. In this respect, this study challenges the use of a small-sized dataset with the order of 3 ($n = 10^3$).

The small-sized real-estate price dataset can incorporate several price feature columns for explaining limited price records. While the increase in the number of price records enhances the prediction accuracy, the increase of influential feature columns reduces the price-prediction ability of the adopted ML approach. The addition of each price feature adds another dimension to the dataset, which complicates the learning of the machine (curse of dimensionality). The issue of high-dimensionality was addressed in literature by eliminating redundant and irrelevant features through various dimensionality reduction techniques (Ayesha et al. 2020): i.e., PCA, factor analysis, forward and backward stepwise regression and linear discriminant analysis (LDA). Reddy et al. (2020) identified PCA as a more effective dimensionality reduction method compared to LDA, when combined with the other ML algorithms, such as DT, SVM, NB and RF. In this regard, the existing studies in the broader literature exercised PCA for the identification and ranking of influential features as well as enhancing the prediction accuracy of ML (Wang and Zhang 2013; Zhan et al. 2021; Wen et al. 2022). However, in the context of real-estate and construction applications, there exists a comparatively small body of literature on AI methods incorporated with PCA. In this regard, the house price and construction dataset was processed using the PCA model combined with radial basis function network (RBFN) (Phan 2019; Xiao and Yan 2019), CNN (Piao et al. 2019), and RF (Piao et al. 2019). Furthermore, SVM and support vector regression (SVR) incorporated with PCA were effectively used within different fields of real-estate and construction applications. The improved performance achieved through the combination of PCA with SVM and SVR has also been well-established in the literature. Son et al., (2012) proposed a PCA-SVR model for cost prediction of commercial buildings, which enhances its performance, compared with the other four analysed data mining methods. All these works have acknowledged the vital role of PCA in dimensionality reduction and consequently the enhanced prediction accuracy of the accompanying algorithm. Despite its efficiency, the topic of the combined performance of PCA with ANN (PCA-ANN) on real-estate and construction was explored only in a few studies (Shi 2009; Xiao and Yan 2019). Shi (2009) used PCA to classify the price determinates for price prediction within the ANN structure. His results showed a good match between PCA-ANN prediction with price labels. Similarly, to predict house prices, Xiao and Yan (2019) combined PCA with RBFN, and their results showed high price-prediction accuracy. In this

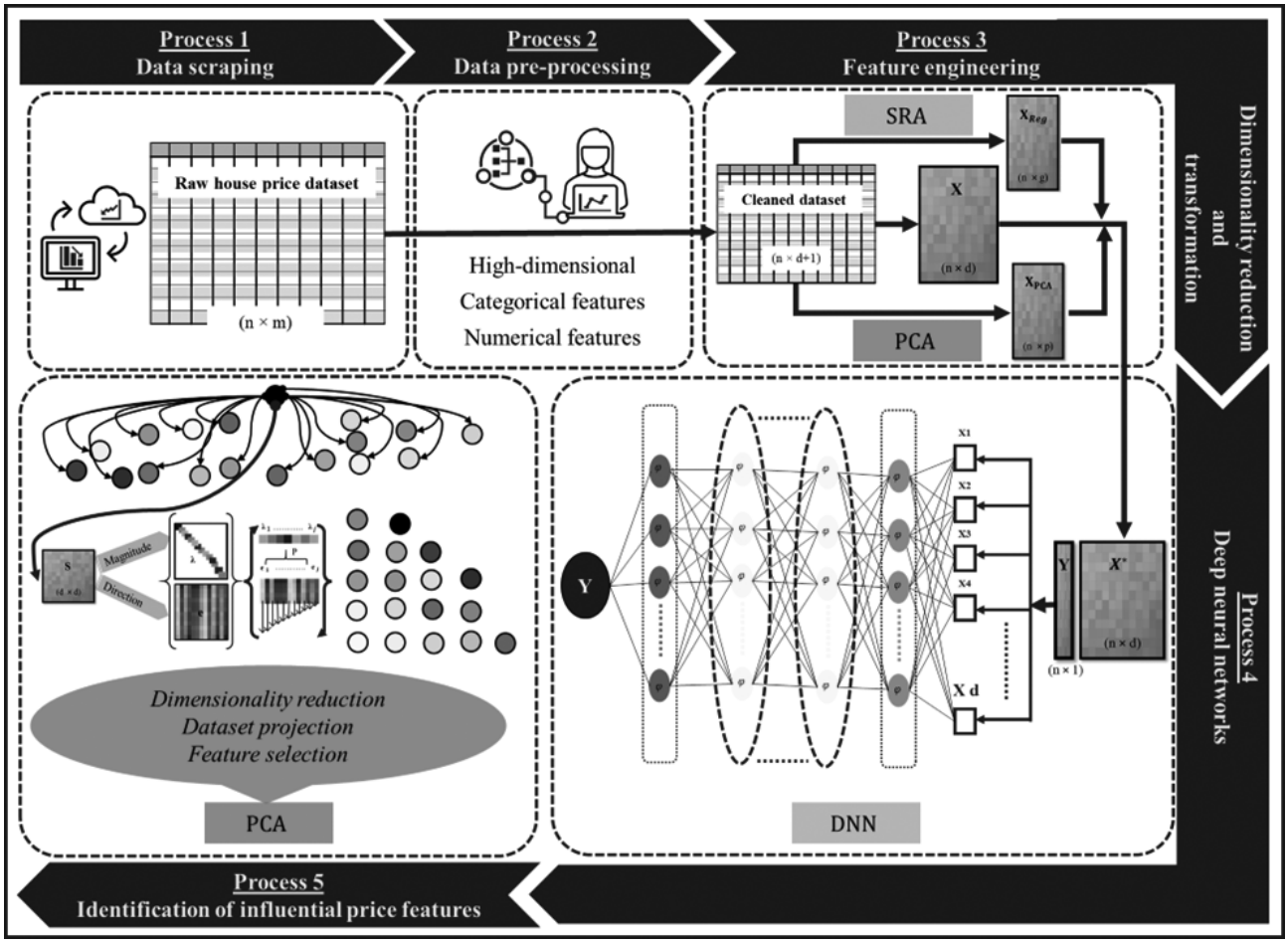

Fig. 1: The methodology followed in this study. DNN, deep neural networks; PCA, principal component analysis; SRA, stepwise regression analysis.

context, the prediction accuracy of DNN and dimensionality reduction of PCA on real-estate price data have not been addressed in the literature. Investigating the accuracy of the combined PCA-DNN model on the high-dimensional dataset still demands further research attention.

To fill in the gap aforementioned, this study purposes a PCA-DNN real-estate price-prediction model that can be utilised on high-dimensional real-estate datasets. Furthermore, the performance of the adopted PCA-DNN model is evaluated against the conventional DNN and SRA-DNN models, concerning both its dimensionality reduction ability and efficiency in handling the encoded categorical dataset. Thus, PCA is expected to improve the DNN model prediction by dimensionality reduction, dataset projection and transformation, and localisation of influential price features. This study includes five main stages: data collection using web scarping, pre-processing, feature engineering, network training and identification of the influential price features with the real-estate dataset (Figure 1).

At the network training stage (Figure 1), to structure the appropriate network architecture of each DNN, SRA-DNN and PCA-DNN models, they are iterated considering different combinations of neurons and layers along with various activation, loss and optimisation functions. This study evaluates both unsupervised and supervised learnings of the models.

3 Research methodology

3.1 Data collection and processing

Real-estate price data, obtained in May 2021 (Sahibinden.com 2021) from an online real-estate advertisement agency in Türkiye, are used in the present study. In this respect, real-estate price records of 42 districts from the central area of Trabzon city in Türkiye are collected. The house price descriptors

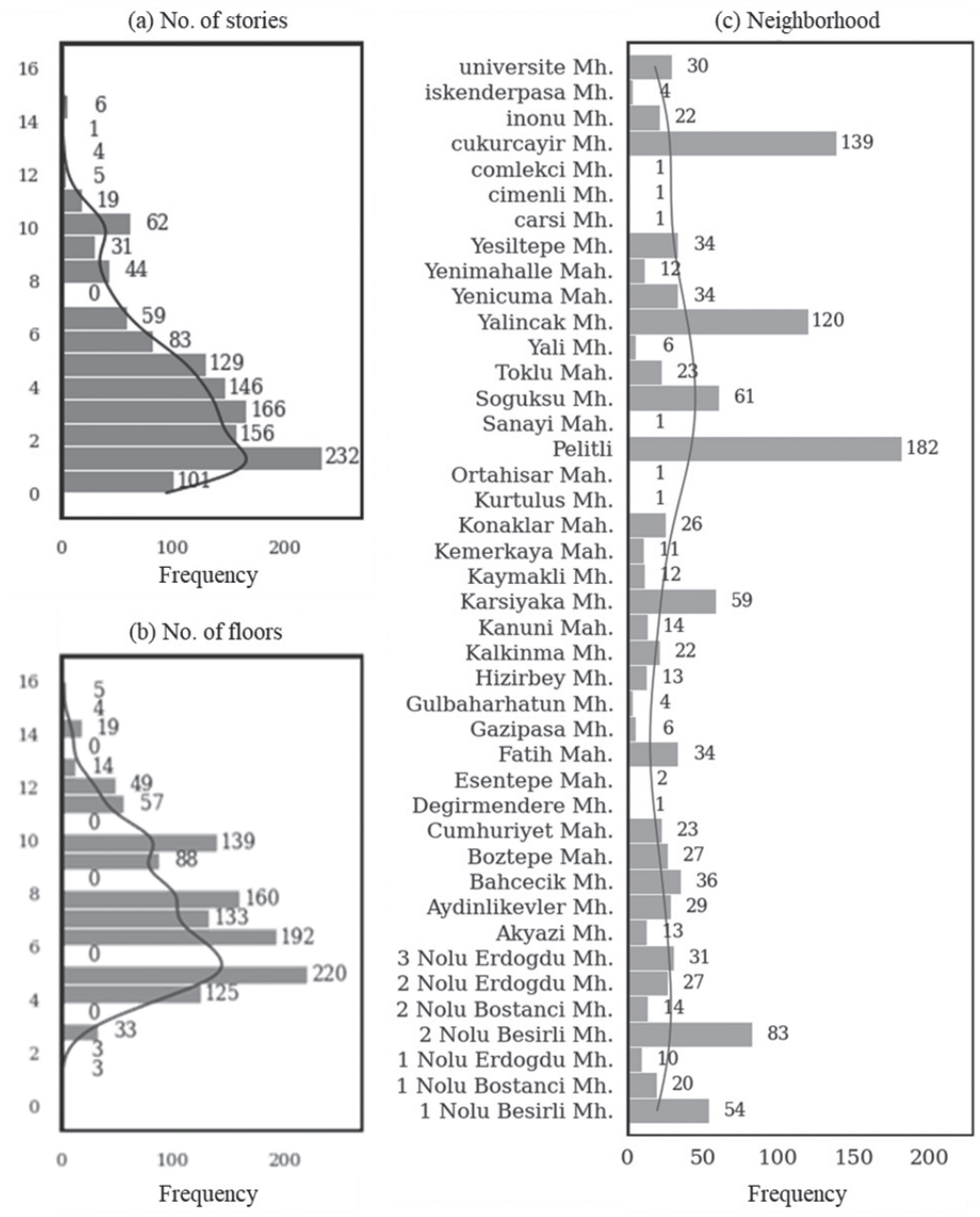

Fig. 2: Frequency of (a) No. of stories, (b) No. of floors and (c) neighbourhood.

on the website such as building age, bedrooms and saloons are used as explanatory price features. This results in a dataset on 1,381 house price records that is explained with 19 feature columns. The data are initially cleaned by the removal of empty and ambiguous data cells. The categorical and numerical features are wrangled into their appropriate data formats, while the noisy data are removed; the relevant details are shown in Figure 3. The cleaned dataset comprised 1,244 price records that were explained by 17 explanatory features. These 17 features includes 7 numerical and 10 categorical features. The seven numerical features are flat area, building age and floor number, number of stories, bathrooms, bedrooms and saloons. The remaining 10 categorical features are further subdivided into six binary and four non-binary nominal categorical features. The six binary categorical features, namely balcony, furniture, amenities, credit availability, video call and swap, are dummy encoded, while the four non-binary categorical features, namely heating, current usage

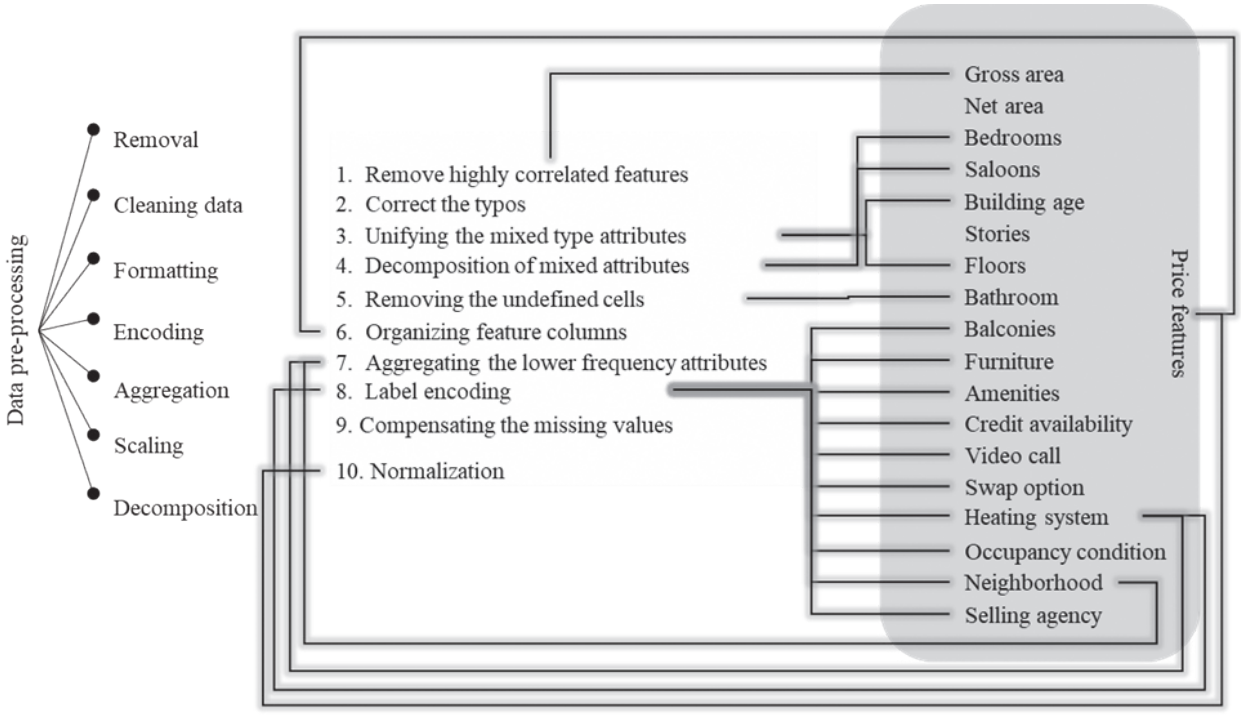

Fig. 3: Data pre-processing stage.

condition, seller and neighbourhood, are encoded using the ordinal encoding method.

Additionally, the real-estate price features obtained from the real-estate project characteristic (PC), along with their attributes, are briefed in Table 1.

It is to be noted that the statistical description for categorical features (Table 1) is comprised of those that are obtained from the dummy encoding procedure.

3.2 Evaluation metrics

To measure the prediction accuracy of these models, three loss metrics (Eqs [1]–[3]) are used: MSE, MAE (Pal 2017) and mean absolute percentage error (MAPE) (Kim and Kim 2016).

$$\mathrm{MAE}=\frac{1}{n}\sum_{i}^{n}\left|y_i^{*}-\widehat{y_i^{*}}\right| \tag{1}$$

$$\mathrm{MAPE}=\frac{1}{n}\sum_{i}^{n}\frac{\left|y_i-\widehat{y_i}\right|}{y_i}\times 100 \tag{2}$$

$$\mathrm{MSE}=\frac{1}{n}\sum_{i}^{n}\left|y_i^{*}-\widehat{y_i^{*}}\right| \tag{3}$$

where n is the number of price records and i is the corresponding record; $\hat{y}_i$ and y_i are the predicted and the actual price values, respectively; and $\widehat{y^{*}}_i$ and y^{*}_i are the normalised predicted and actual price values associated with each record i, respectively. Notably, MAPE is sensitive to normalisation and therefore is calculated over unnormalised price data. In contrast to MAPE, both MSE and MAE metrics are sensible to normalisation scaling and therefore are used over the normalised predicted and actual price labels.

3.3 Dimensionality reduction and feature selection

In this paper, simultaneous to dimensionality reduction ability, PCA is used to enhance the quality of encoded price data. Therefore, firstly the dimensionality reduction ability and feature selection quality of PCA are compared with SRA. Secondly, DNN models are trained with PCA-extracted features and SRA-selected features. Moreover, in conducted analyses, NumPy (Harris et al. 2020) and Pandas (McKinney 2010) Python libraries were used, respectively, for managing matrices and arrays, and data manipulation and analysis. Outcome of the pre-processed dataset is presented in a feature matrix, as seen in Eq. (4).

Tab. 1: PC attributes – numerical and categorical labels and frequencies.

Identifier	Feature name	Mean	Standard deviation	Feature type	Feature attributes	Frequency of attribute
PC1	Area	313.88	105.94	Numerical	(89, 286]	497
					(286, 482]	664
					(482, 678]	74
					(678, 874]	7
					(874, 1070]	2
PC2	Room	3.08	0.86	Numerical	1 bedroom	45
					2 bedrooms	168
					3 bedrooms	776
					4 bedrooms	166
					5 bedrooms	79
					6 bedrooms	9
					7 bedrooms	1
PC3	Saloon	1.04	0.20	Numerical	0 saloon	3
					1 saloon	1,191
					2 saloons	50
PC4	Building age	11.14	11.95	Numerical	0–4 years	595
					5–10 years	218
					11–15 years	163
					16–20 years	135
					21–25 years	71
					26–30 years	38
					≥31 years	24
PC5	No. of stories	7.27	2.66	Numerical	1–15 stories	Illustrated in Figure 2a.
PC6	Floor No.	3.90	3.05	Numerical	−1 to 15 floor	Illustrated in Figure 2b.
PC7	No. of bathrooms	1.61	0.59	Numerical	1 bathroom	549
					2 bathrooms	635
					3 bathrooms	54
					4 bathrooms	6
PC8	Balconies	0.05	0.22	1	With balcony	1,182
				2	Without balcony	62
PC9	Furniture	0.97	0.18	1	Furnished	42
				2	Not furnished	1,202
PC10	Amenities	0.68	0.46	1	Amenities included	394
				2	Amenities not included	850
PC11	Credit availability	0.10	0.30	1	Available	1,118
				2	Unavailable	126
PC12	Video call	0.53	0.50	1	Available	589
				2	Unavailable	655
PC13	Swap option	0.84	0.37	2	Ready for swap	200
					No swap	1,044
PC14	Heating system	0.78	1.66	1	Natural gas	1,006
				2	Central	173
				3	gas stove	6
				4	Air conditioning	6
				5	Stove	29
				6	Underfloor heating	17
				7	Fireplace	7
PC15	Occupancy condition	0.54	0.80	1	Unoccupied	812
				2	Occupied by owner	243
				3	Under rent	189
PC16	Selling agency	0.18	0.46	1	Real-estate agent	1,052
				2	Construction company	36
				3	Private owners	156
PC17	Neighbourhood			42	1–42 districts	Illustrated in Figure 2c.

PC, project characteristic.

This dataset includes n = 1,244 price observations and d = 17-dimensional feature spaces.

$$\mathbf{X} = \begin{bmatrix} x_{11} & \cdots & x_{1d} \\ \vdots & \ddots & \vdots \\ x_{n1} & \cdots & x_{nd} \end{bmatrix} \tag{4}$$

$$x_{ij} \in R^{n \times d},\ \forall\ i = 1, 2, \ldots,\ n,\ \forall\ j = 1, 2, \ldots,\ d$$

$$\mathbf{Y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}$$

$$y_i \in R^n,\ \forall\ i = 1, 2, \ldots,\ n$$

where $\mathbf{X}$ is a matrix of size ($n \times d$), and n is the total number of price records (samples). Each of these price records has d number of real-estate–specific features (target factors, x_j), while the corresponding matrix $\mathbf{X}$ is constituted with observation values, x_{ij}. Correspondingly, $\mathbf{Y}$ is a vector size n, associated with 1,244 price records. Then, the cleaned dataset is normalised through Eq. (5).

$$x_{ij}^{*} = \frac{x_{ij} - \mu_j}{\sigma_j} \tag{5}$$

where x_{ij} and x_{ij}^{*} represent i^{st} price record associated with j^{st} feature and its normalised value, respectively. Also, μ_j and σ_j are the mean and standard deviation, related to all records in each feature space calculated using Eqs (6) and (7).

$$\mu_j = \frac{1}{n} \sum_{i=1}^{n} x_{ij} \tag{6}$$

$$\sigma_j = \left[\frac{1}{n-1} \sum_{i=1}^{n} \left(x_{ij} - \mu_j \right)^2 \right]^{1/2} \tag{7}$$

To avoid the zero-division error, corresponding standard deviations with 0 instances are replaced with 1. Consequently, the normalised data becomes unit-free while preserving the original data covariance.

3.3.1 Principal component analysis

For the computation of principal components, the steps below are followed.

Step 1: Data normalisation
To centre the dataset around the origin, before projecting data into a lower-dimensional subspace, data are normalised as seen earlier in Eq. (5).
Step 2: Computation of covariance matrix
The covariance matrix is the measurement of the spread of each feature dimension away from the mean concerning the other feature dimensions. The covariance matrix $\mathbf{S}$ is obtained using equation Eq. (8).

$$\mathbf{S} = \begin{bmatrix} s_{11} & \cdots & s_{1d} \\ \vdots & \ddots & \vdots \\ s_{d1} & \cdots & s_{dd} \end{bmatrix} \tag{8}$$

$$s_{kr} \in R^{d \times d},\ \forall\ k = 1, 2, \ldots, d,\ \forall\ r = 1, 2, \ldots, d$$

where $\mathbf{S}$ is a ($d \times d$) matrix that each of its indices $\boldsymbol{s_{kr}}$ stands for in the correlation coefficient. $\boldsymbol{s_{kr}}$ is calculated using Eq. (9).

$$s_{kr} = \frac{1}{n} \sum_{i=1}^{n} x_{ik}^{*} x_{ir}^{*} \tag{9}$$

Step 3: Calculation of eigenvalues and eigenvectors.
For calculation of eigenvalues, Eq. (10) is to be solved for λ correlated with covariance matrix $\mathbf{S}$.

$$\mathbf{S}\boldsymbol{e}_{ij} = \boldsymbol{\lambda}\boldsymbol{e}_{ij} \tag{10}$$

where λ and $\boldsymbol{e}_{ij}$ are eigenvalues and eigenvectors, respectively. Eigenvectors are associated with the direction of uncorrelated features in the dataset while eigenvalues stand for the information retained by each feature. Eigenvalues and eigenvectors obtained upon the solution of Eq. (10) are to be sorted in ascending order, based on the order of eigenvalues. Moreover, to determine how many features of a dataset are kept while projecting the dataset into low subspace, Eq. (11) is executed.

$$\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_k$$

$$X_k = \sum_{r=1}^{d} e_{rk}\, x_{ij}^{*} \tag{11}$$

$$\breve{\mathbf{X}} = \begin{bmatrix} \vdots & \vdots & \vdots \\ X_1, & X_2 \ldots, & X_k \\ \vdots & \vdots & \vdots \end{bmatrix}$$

where X_k is a k^{th} principal component derived using sorted eigenvector e_{rk} multiplied by normalised sample vector x_{ij}^{*}. Also, $\breve{\mathbf{X}}$ is an $(n \times k)$ matrix generated from all the project features into the new subspace. Here, to obtain the overall data variance, the accumulated contribution of these principal components is to be selected as follows in Eq. (12).

$$\mathbf{X}_{\mathbf{PCA}} \subseteq \breve{\mathbf{X}}$$

$$X_{PCA} \in R^{n \times P}, \ \forall \ p < k \tag{12}$$

where $\mathbf{X}_{\mathrm{PCA}}$ is an $(n \times p)$ matrix, generated by selecting the $p < k$ principal components. In other words, the original $(n \times d)$ data dimension at this step is reduced to $(n \times p)$ dimension.

3.3.2 Deep neural network

The procedure to structure a DNN model is outlined in Figure 4.

As depicted in Figure 4, the first step is to initialise the parameters of the model referred to as hyperparameters. The main hyperparameters of a DNN model are learning rate, activation function, number of epochs, layers and neurons, as well as activation, loss and optimisation functions. Hence, the first layer of the network is initialised using the normalised feature matrix $\mathbf{X}^{*}$, while assigning the weight and bias parameters randomly, as indicated in Eq. (13).

$$\mathbf{Z}_1 = \mathbf{W}_1 \mathbf{X}^{*} + \mathbf{b}_1 \tag{13}$$

where $\mathbf{W}_1$ and $\mathbf{b}_1$ are the weight and bias matrices of the first layer of DNN. Here, $\mathbf{Z}_1$ is the pre-activation parameter, which is the resultant input of the activation function. Furthermore, the initialised parameters in Eq. (13) are used to compute the activation function as illustrated in Eq. (14).

$$\mathbf{A}_1 = \boldsymbol{\varphi}\left(\mathbf{Z}_1\right) \tag{14}$$

where $\mathbf{A}_1$ is the activation of the first layer. Here, $\varphi(.)$ is the activation function. In this study, relu, softmax, softsign and tanh activation functions were iterated to find the optimum activation of the hidden layers. For the final layer, a linear activation function is used. Accordingly, using the forward propagation, the extracted information is transferred to the next layer, as shown in Eqs (15) and (16).

$$\mathbf{Z}_L = \mathbf{W}_L \ \mathbf{A}_{L-1} + \mathbf{b}_L \tag{15}$$

$$\mathbf{A}_L = \boldsymbol{\varphi}\left(\boldsymbol{Z}_L\right) \tag{16}$$

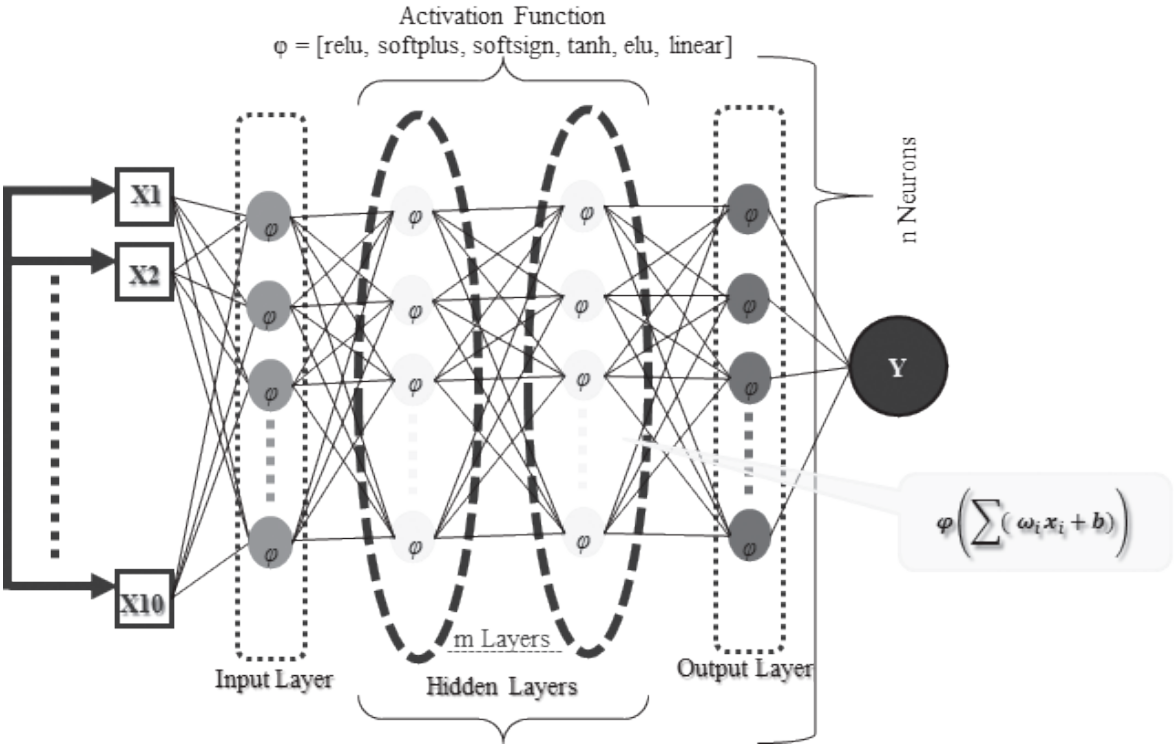

Fig. 4: Outline of DNN procedure. DNN, deep neural networks.

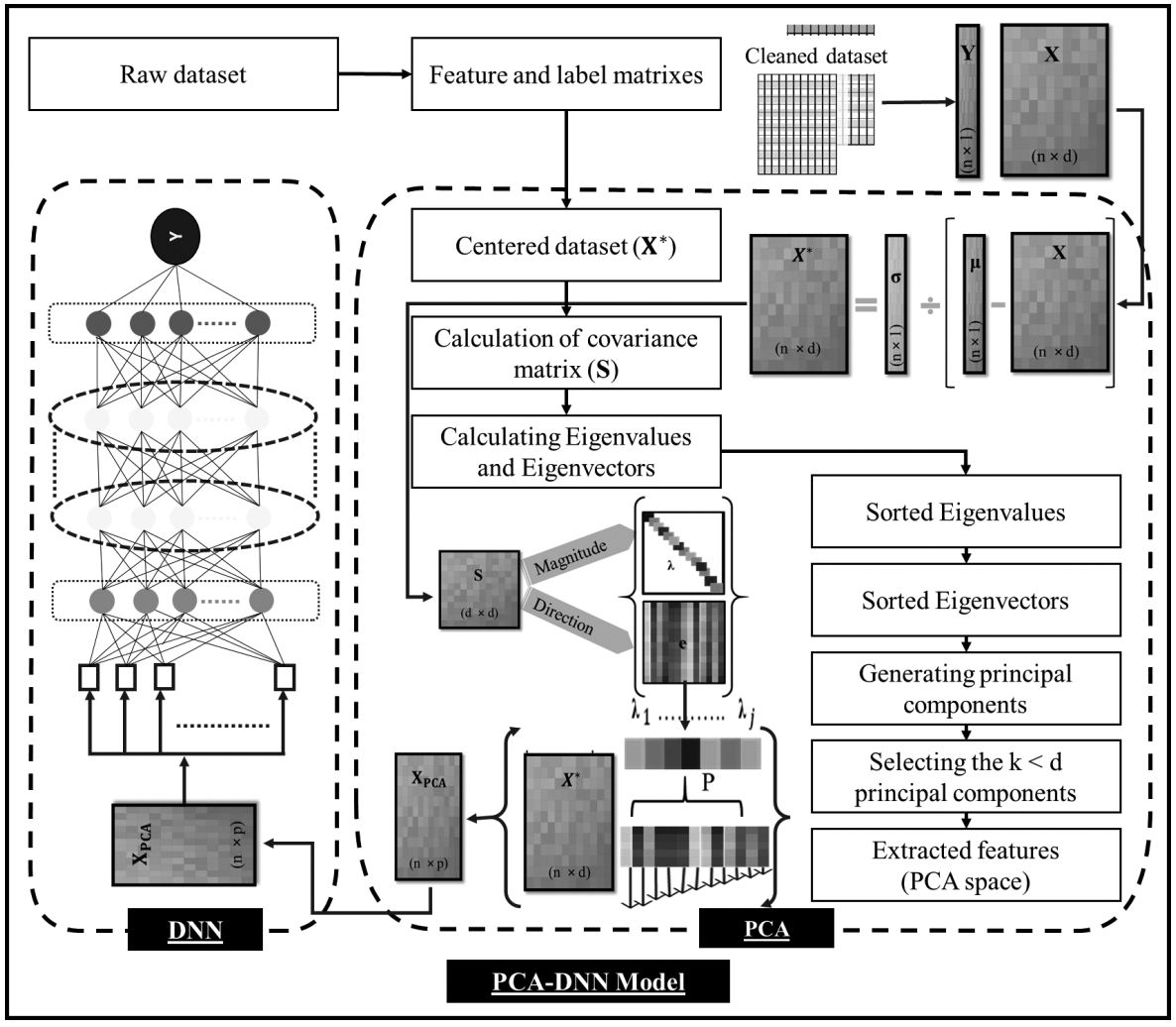

Fig. 5: PCA-DNN model. DNN, deep neural networks; PCA-DNN, principal component analysis-deep neural networks; PCA, principal component analysis.

where $\mathbf{A}_L$ is the activated function from the earlier layer, $\mathbf{W}_L$ is weight matrix and $\mathbf{b}_L$ is bias. Further, L is the number of the associated layer. After the calculation of the activation function in each layer, the cost function is to be computed using Eq. (17).

$$J = -\frac{1}{n}\sum_{i=1}^{n}\left(\hat{\mathbf{Y}}_i \ln\left(A_{L_i}\right) + \left(1+\widehat{\mathbf{Y}}_i\right)\ln\left(1-A_{L_i}\right)\right) \tag{17}$$

In Eq. (17), J is the average value of cross-entropy cost, while $\widehat{\mathbf{Y}}_i$ is the predicted price label. Through a procedure called backward propagation, the calculated cost function returns to the initial layers to improve the weights. Afterwards, the backward propagation is used to compute the gradient of the calculated loss function for the adjusted parameters. Here, the adjusted weight and bias are updated using the mathematical expression given in the following equations, Eqs (18)–(20).

$$\frac{\partial J}{\partial \mathbf{W}_L} = \frac{1}{n}\left(\frac{\partial L}{\partial Z_L}\,\mathbf{A}_{L-1}{}^T\right) \tag{18}$$

$$\frac{\partial J}{\partial b_L} = \frac{1}{n}\sum_{i=1}^{n}\frac{\partial L}{\partial Z_{L_i}} \tag{19}$$

$$\frac{\partial L}{\partial \mathbf{A}_{L-1}} = \mathbf{W}_L{}^T\frac{\partial L}{\partial Z_L} \tag{20}$$

where $\frac{\partial J}{\partial \mathbf{W}_L}$ and $\frac{\partial J}{\partial b_L}$ are derivatives of wight $\mathbf{W}_L$ and bias $\mathbf{b}_L$ matrices; and $\frac{\partial L}{\partial \mathbf{A}_{L-1}}$ and $\frac{\partial L}{\partial Z_L}$ are derivatives of the activated layer $\mathbf{A}_{L-1}$ and pre-activation $\mathbf{Z}_L$, respectively.

Furthermore, using the calculated gradient descent from the previous step, the weight and bias parameters are to be updated as stated in Eqs (21) and (22).

$$\mathbf{W}_L = \mathbf{W}_L + \alpha \frac{\partial J}{\partial \mathbf{W}_L} \tag{21}$$

$$\mathbf{b}_L = \mathbf{b}_L + \alpha \frac{\partial J}{\partial \mathbf{b}_L} \tag{22}$$

where α is the learning rate. Afterwards, the decribed procedures are to be repeated for the predetermined numbers of iterations. To prevent DNN from overfitting, the patience value is set as 10, which stops the model 10 steps after reaching its optimum loss value. Finally, the trained parameters along with hyperparameters are used for label prediction. The described DNN procedure is to be repeated over $\mathbf{X}_{Reg}$ selected and $\mathbf{X}_{PCA}$ extracted features, as detailed in the following section.

4 Price-prediction models

4.1 Model development

In this work, a hybrid PCA-DNN model, using 1,244 extracted feature records, $\mathbf{X}_{PCA}$ (Eq. [12]), is adopted over small-sized real-estate price data (Figure 5).

The PCA-DNN model extracts and translates real-estate price features into new feature space, significantly enhancing the quality of feature attributes as well as the prediction ability of the DNN model.

4.2 Benchmark model

For performance evaluation of the developed model, backward SRA is also used. SRA selects influential

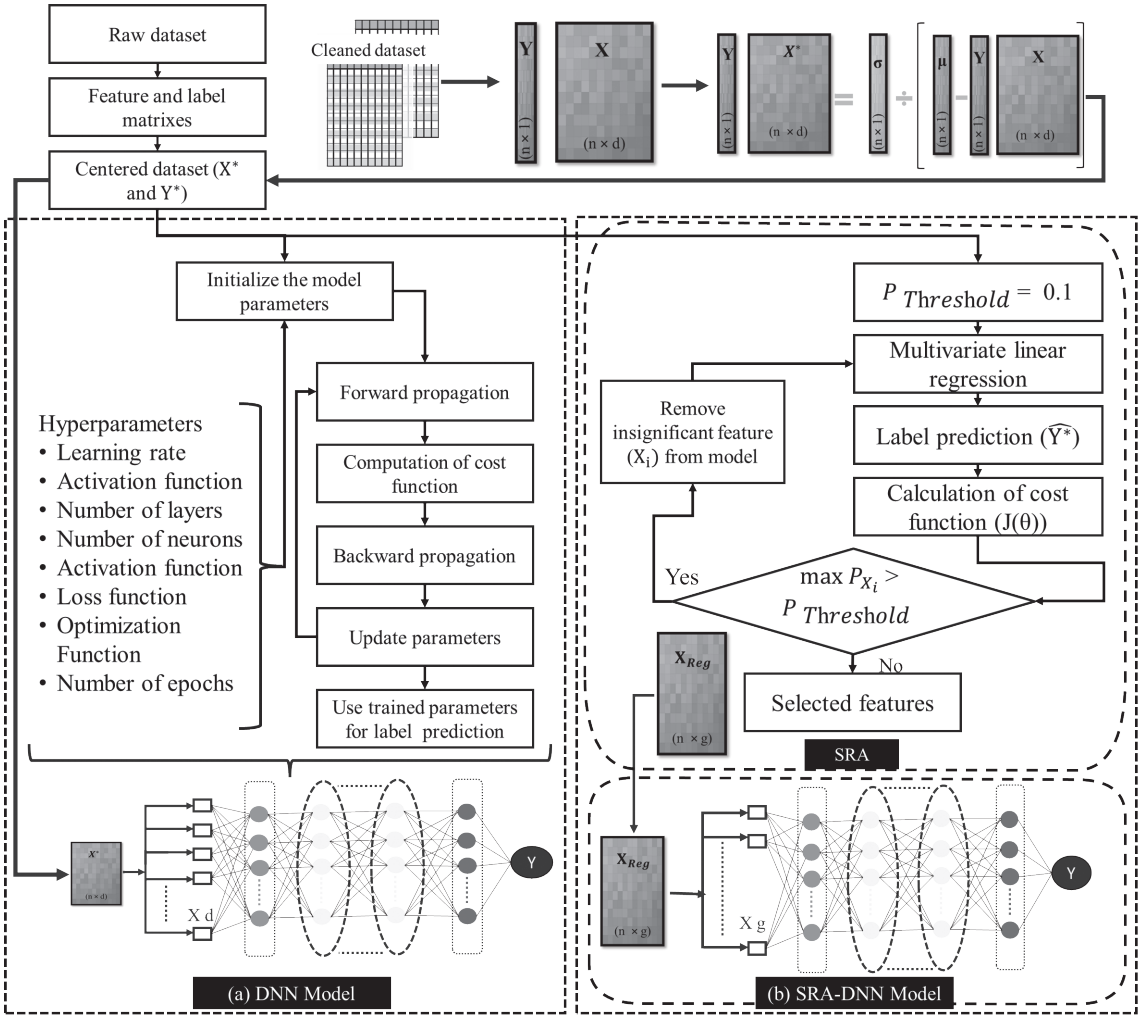

Fig. 6: Structure of (a) DNN and (b) SRA-DNN models. DNN, deep neural networks; SRA-DNN, stepwise regression analysis-deep neural networks.

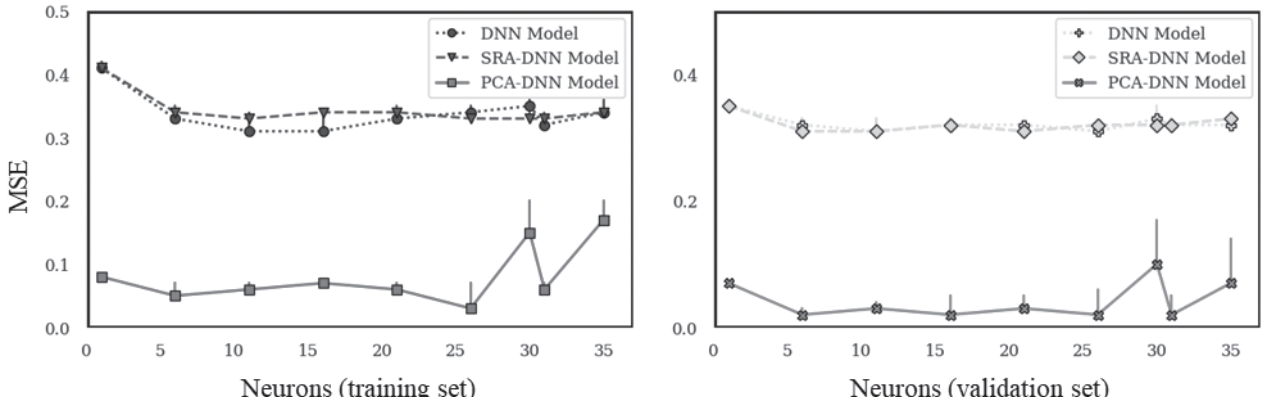

Fig. 7: Training and validation of MSE for the different number of neurons, in supervised learning scenarios of DNN, SRA-DNN and PCA-DNN models. DNN, deep neural networks; MSE, mean square error; PCA-DNN, principal component analysis-deep neural networks; SRA-DNN, stepwise regression analysis-deep neural networks.

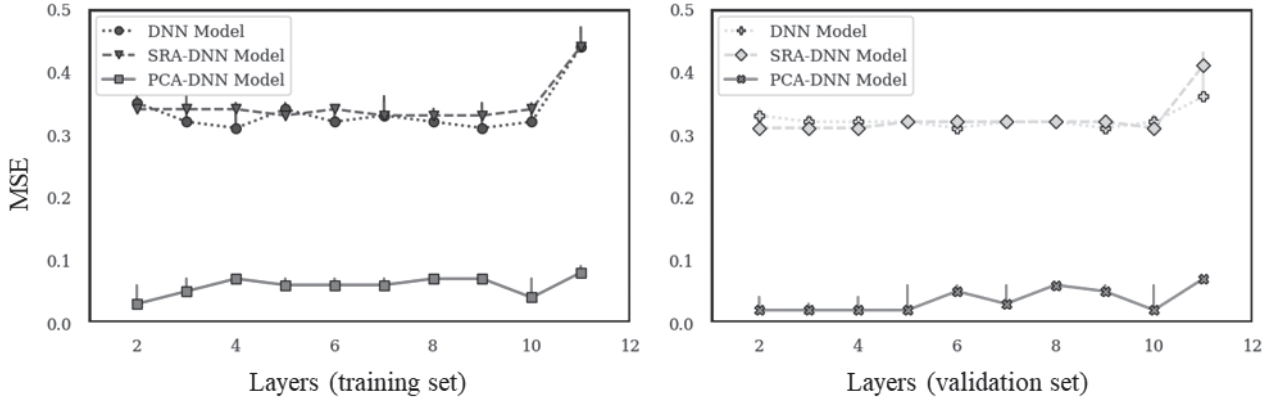

Fig. 8: Training and validation of MSE for the different number of layers, in supervised learning scenarios of DNN, SRA-DNN and PCA-DNN models. DNN, deep neural networks; MSE, mean square error; PCA-DNN, principal component analysis-deep neural networks; SRA-DNN, stepwise regression analysis-deep neural networks.

Tab. 2: The optimum network architecture of the three selected models.

Factors	DNN Model	SRA-DNN Model	PCA-DNN Model
Number of neurons	30	20	30
Number of neurons (output layer)	20	20	20
Number of layers	5	5	5
Total trainable parameters	2,061	1,921	2,041
Activation function (output layer)	Linear	Linear	Linear
Activation function	relu	relu	relu
Optimisation function	Adam	Adam	Adam
Loss function	mse	mse	mse
Number of features	17	10	15

DNN, deep neural networks; PCA-DNN, principal component analysis-deep neural networks; SRA-DNN, stepwise regression analysis-deep neural networks.

Tab. 3: Performance of selected optimum DNN, SRA-DNN and PCA-DNN models.

Model	Wall times	CPU time	Epoch	MAE	MAPE	MSE
DNN	3.02 s	3.32 s	20	0.43	27%	0.42
SRA-DNN	6.06 s	7.05 s	30	0.42	22%	0.39
PCA-DNN	6.27 s	7.27 s	160	0.23	14%	0.10

DNN, deep neural networks; MAE, mean absolute error; MAPE, mean absolute percentage error; MSE, mean square error; PCA-DNN, principal component analysis-deep neural networks; SRA-DNN, stepwise regression analysis-deep neural networks.

features through forward and backward stepwise analyses. Among these, backward SRA uses all the variable features within a single model and comparatively retains a larger value of R^2. Hence, SRA attempts to select a

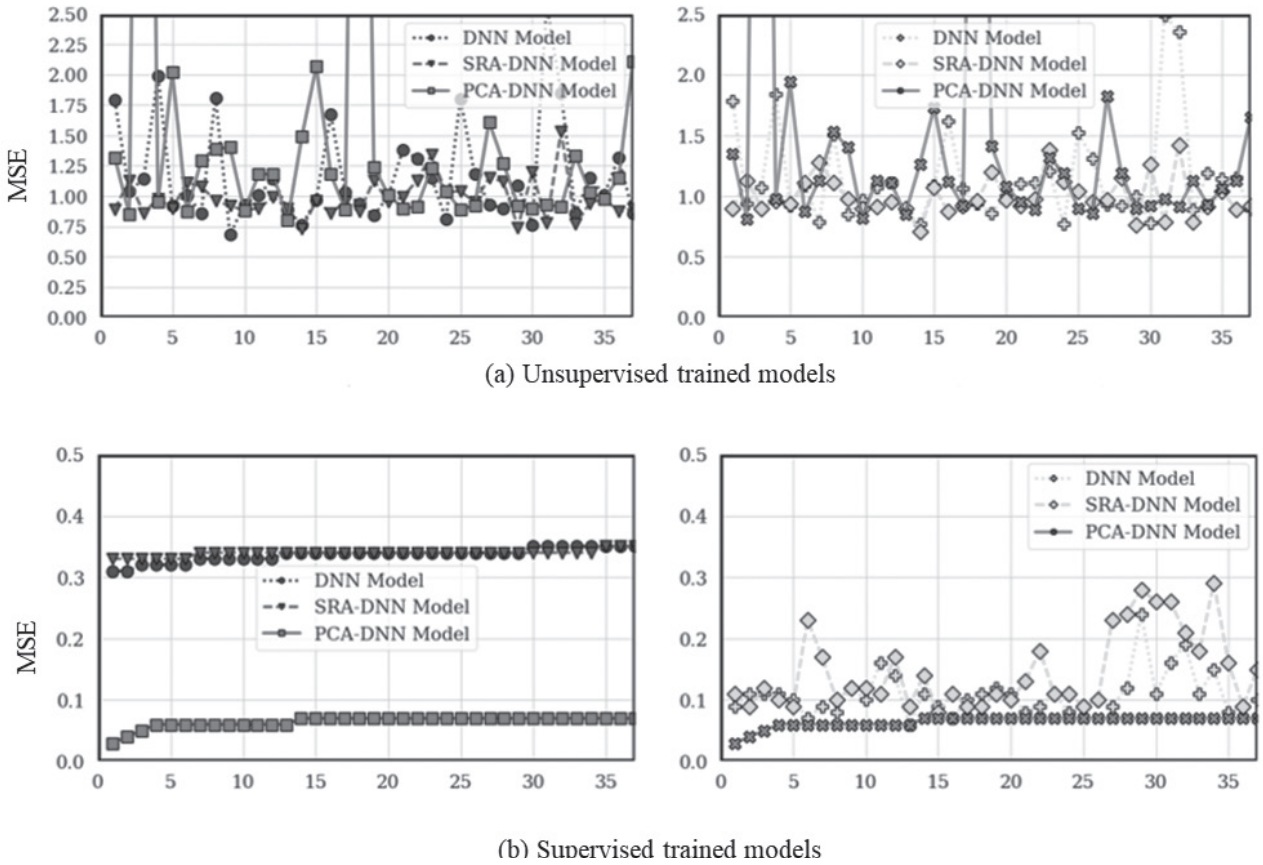

Fig. 9: Unsupervised (a) and supervised (b) training and validation of MSE values for best performing DNN, SRA-DNN and PCA-DNN models. DNN, deep neural networks; MSE, mean square error; PCA-DNN, principal component analysis-deep neural networks; SRA-DNN, stepwise regression analysis-deep neural networks.

minimum number of independent variables (feature columns) with the most meaningful explanation of the dependent variable (price records). The backward SRA initiates with the evaluation of all explanatory features, **X** (Eq. [4]), with respect to their probability of occurrence in sample data, called *P*-value or significance level. Afterwards, the selected features, $\mathbf{X}_{Reg}$, are obtained and fed into the DNN model.

Therefore, this study compares the predictive and computational cost performance of the adopted PCA-DNN model with DNN and SRA-DNN models to evaluate its efficiency in the handling of the high-dimensional dataset (Figure 6).

First, the PCA-DNN model's ability in handling categorical real-estate data is to be compared with the DNN model (Figure 6a), using all of its principal components. Likewise, the PCA-DNN model is further compared with the SRA-DNN model (Figure 6b) to consider its dimensionality reduction ability over the high-dimensional categorical real-estate dataset.

4.3 Model training

Selection of the optimum network architecture plays a crucial role in price-prediction performance, and thus the optimal network architecture for each of the three models is obtained for the respective analyses. Upon completion of each iteration step, the loss function is computed for evaluation. Amongst the iterated loss functions, MAE (Eq. [1]) and MSE (Eq. [3]) show the lowest average error, and therefore MSE is selected, as the loss metric, for all the three models. In this work, different optimisation functions have been taken into consideration, whereby Adam and Nadam optimisations exhibited the best performance, and thus Adam optimisation is used for all models. Also, DNN-based models are iterated through conventional regression activation functions, namely Tanh, Softsign, Softplus, Relu and Elu, whereby the Relu activation function is selected for all three models. The observed MSE values against the different numbers of neurons are evaluated

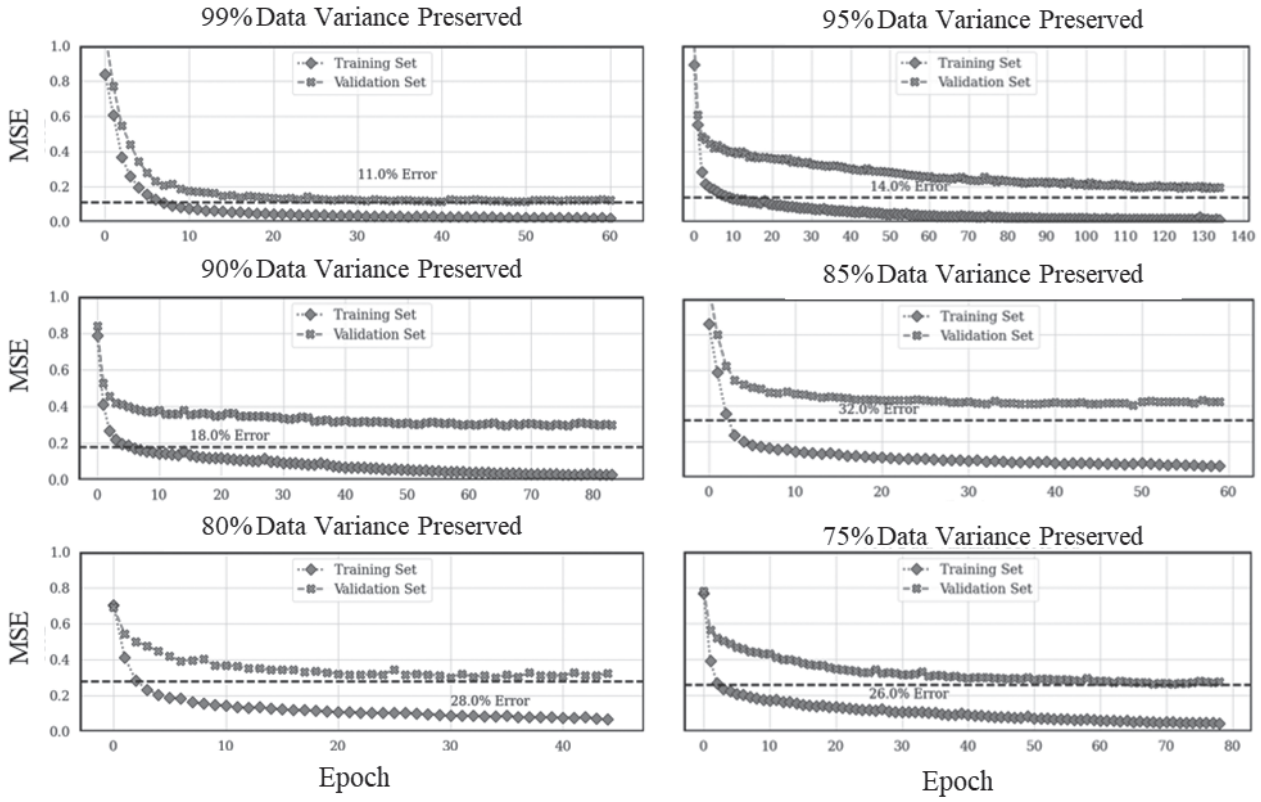

Fig. 10: Contribution of number of principal components in error obtained in PCA-DNN model. MSE, mean square error; PCA-DNN, principal component analysis-deep neural networks.

for the three models (Figure 7). The marked values in Figure 7 show the observation average throughout the performed iterations.

Based on Figure 7, the increase in the number of neurons improves the training and validation errors, while all models reach their best performance between 20 and 30 neurons. In supervised learning scenarios, the PCA-DNN model outperforms DNN and SRA-DNN models. Some of the architectural configurations of PCA-DNN models yield up to 98% prediction accuracy, whereas the accuracy of the other models does not surpass the limit of 75%. Moreover, Figure 8 displays the behaviour of three DNN-based models concerning the different number of layers. The marked values in Figure 8 show the observation average throughout the performed iterations.

Based on Figure 8, in the supervised scenario, all models are best trained with 5–10 layers. However, after 10 layers, the models are not trained and the MSE values increase for all three models. Accordingly, all models are trained using six layers.

Afterwards, the trained parameters along with hyperparameters are used for label prediction. Each of the DNN, SRA-DNN and PCA-DNN models is iterated over different parameters to find the best model configuration with the best price-prediction accuracy. Accordingly, the optimum hyperparameters have been adjusted for each of the three models, as presented in Table 2.

Table 2 shows the best configuration for all models, and therefore the efficiency of the PCA-DNN model can be compared with the best possible performance of the DNN and SRA-DNN models.

5 Results and discussions

5.1 Model evaluation

Based on this, the optimal model architecture is adjusted for each of the three selected models, and the results are presented in Table 3.

Table 3 highlights the improvement in prediction accuracy from 40% MSE (DNN model) to 10% MSE (PCA-DNN model). This demonstrates the PCA-DNN model's ability in enhancing the quality of the encoded categorical datasets, together with simultaneously reducing the high-dimensionality of the sparse real-estate dataset. This finding is also ascertained by comparing the

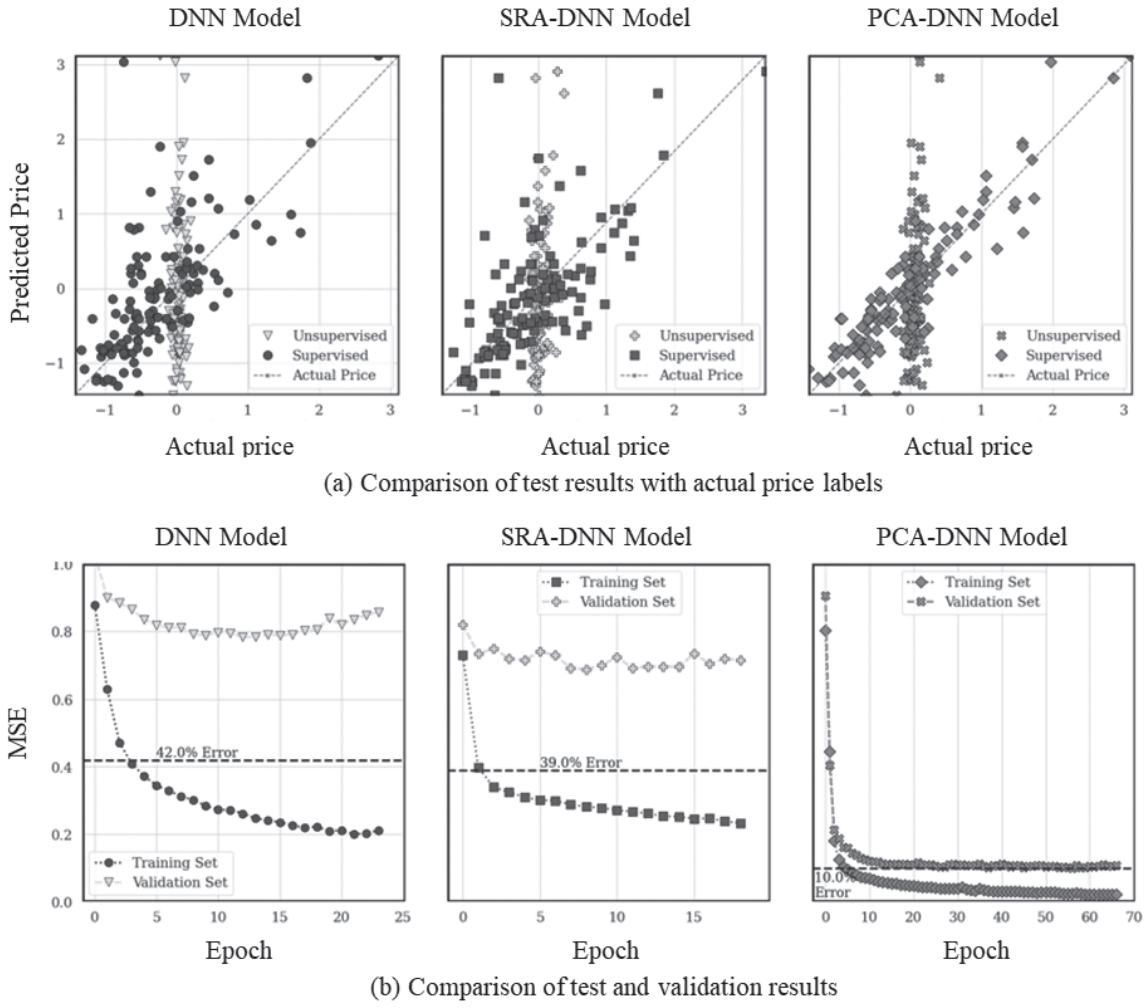

Fig. 11: Learning pattern for unsupervised learning and supervised learning (a) and training and validation accuracy (b) for DNN, SRA-DNN and PCA-DNN models. DNN, deep neural networks; PCA-DNN, principal component analysis-deep neural networks; SRA-DNN, stepwise regression analysis-deep neural networks.

maximum prediction accuracy of the adopted PCA-DNN model with those of the DNN and SRA-DNN models. In the case of both DNN and SRA-DNN, the accuracy could not exceed 60%. As a result, the accuracy obtained by this study over the order 3 ($n = 10^3$) dataset (14% MAPE) could outperform those achieved using DNN in a study by Seya and Shiroi (2021) over the order 4 ($n = 10^4$) dataset (21% MAPE).

Besides, the unsupervised and supervised performance of all models within both training and validation sets are compared, whereby the best performing DNN, SRA-DNN and PCA-DNN models with the least MSE values are shown in Figure 9.

In all iterations, the differences between the training and evaluation MSE values for DNN and SRA-DNN models are larger than the observed difference in MSE values from the PCA-DNN model (Figure 9). The PCA-DNN model achieved an average of 5%–10% MSE contrasted to the minimum of 30%–35% MSE obtained by the other two models. The iteration results demonstrate the superior performance of the PCA-DNN model compared with the DNN and SRA-DNN models, in both unsupervised and supervised configurations.

Moreover, the percentage of data variance kept for the selected principal components directly affects the model performance, as shown in Figure 10.

For computational efficiency, the PCA-DNN models can be trained with fewer components (Figure 10), while still yielding better performance than the DNN and SRA-DNN models. However, the model characterised by the best accuracy is used for comparison, namely the PCA-DNN model with 15 principal components, which preserves 99% of data variance. Additionally, the model comparison demonstrates the efficiency of the

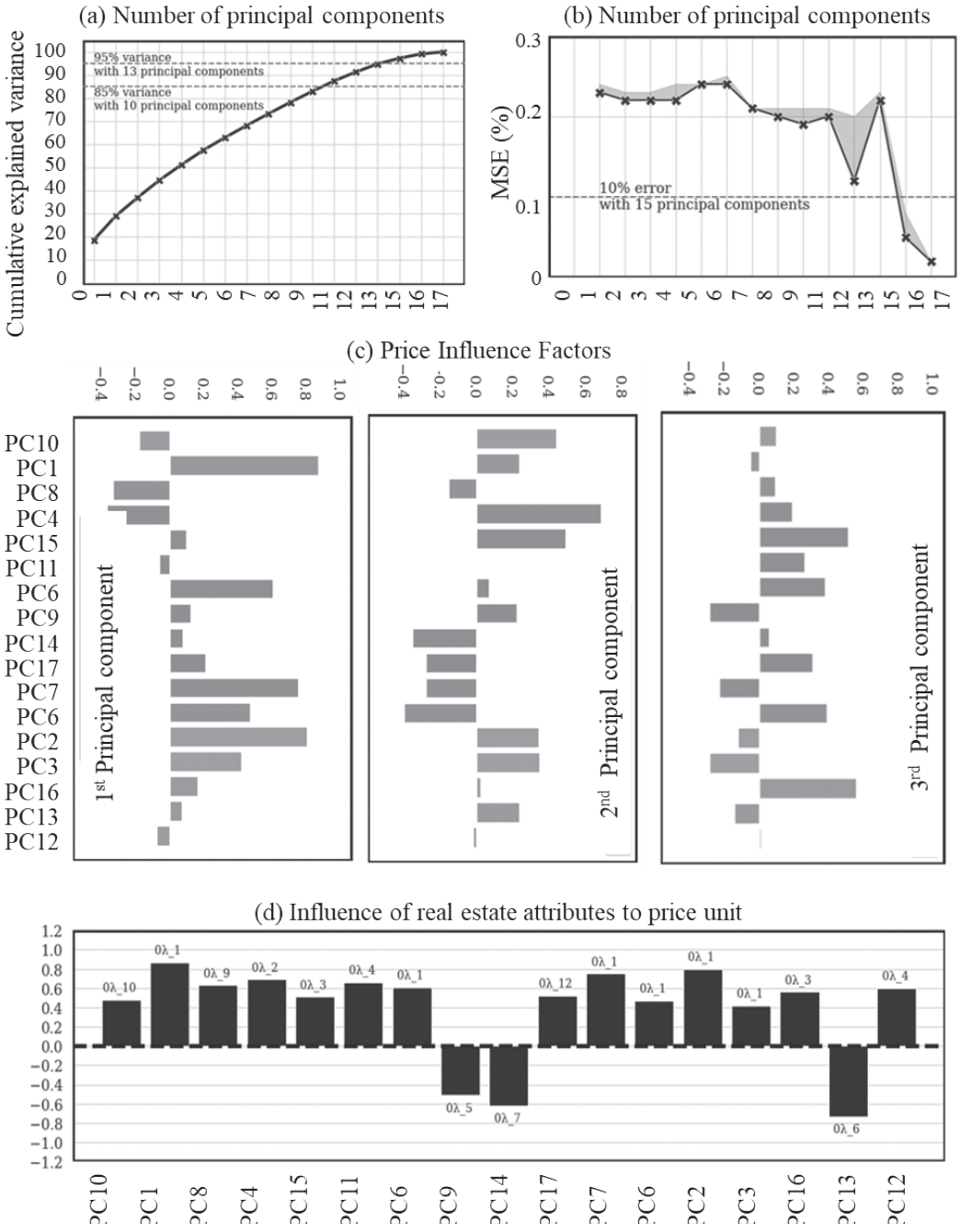

Fig. 12: Total variance preserved by principal components (a). MSE values were obtained for selected principal components (b). Detailed influence of first three principal components (c). Influence of all principal components (d). MSE, mean square error; PCs, project characteristics.

PCA-DNN model in improving the prediction accuracy of both unsupervised and supervised learning scenarios (Figure 11). The adoption of the PCA-DNN model compensates for the reduced prediction accuracy due to the limited number of price records and the categorical nature of feature columns.

Both DNN and SRA-DNN models reached their optimum performance at initial epochs, unlike the proper training observed in the PCA-DNN model throughout the greater number of epochs (Figure 11). The observed training losses for DNN and DNN-SRA models decrease with the rise in validation losses, which implies the overfitting problem and training difficulty (Figure 11). By contrast, the accuracy of the PCA-DNN model reaches over 90% without any overfitting issues. The extracted data within each principal component contain the variance in

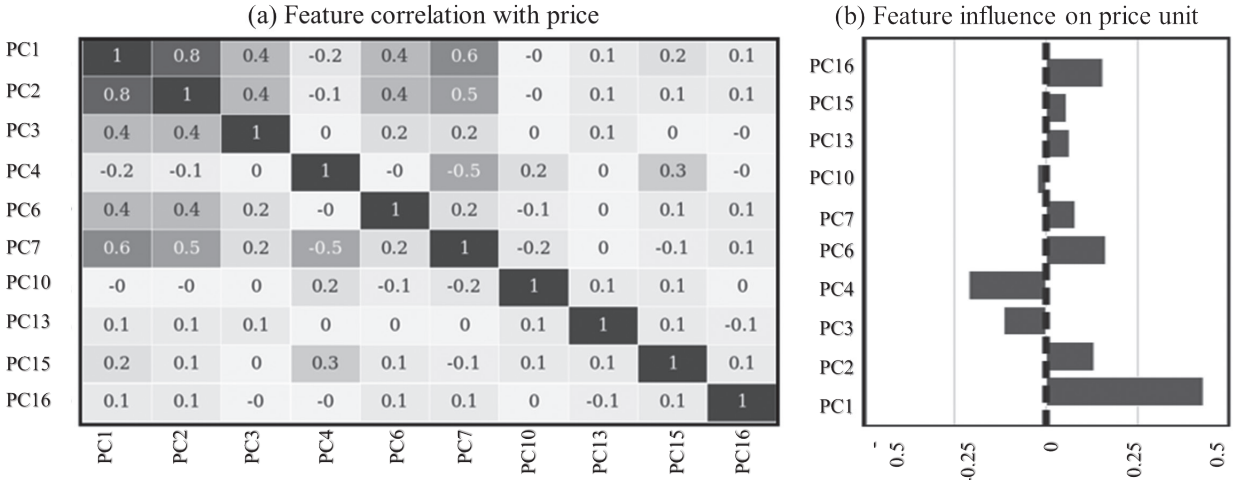

Fig. 13: Feature correlation with price (a). Feature influence on price unit (b). PC, project characteristic.

numerical and categorical price feature attributes in continuous numerical format, which facilitates the learning procedure of the neurons within each layer.

5.2 Identification of influential price features

In addition to dimensionality reduction, PCA is used to find the influential feature attributes (Figure 12). The number of selected principal components determines the cumulative variance preserved in a dataset (Figure 12a). Additionally, Figure 12b shows the relationship between the number of principal component and MSE values obtained from the PCA-DNN model. The PCA localises the crucial features with the most impact on the house price. The first three principal components are responsible for 37% of the variance in the dataset, and their contribution to explaining the price units is detailed in Figure 12c. The 1st principal component is associated with the spatial attributes, namely area, number of stories, floor number, bathrooms, bedrooms and saloons (Figure 12c). The age attribute of the building is shown by the 2nd principal component. Similarly, the 3rd principal component is responsible for usage conditions and the seller agency of the flat (Figure 12c). The other principal components are responsible for relatively smaller data variance, while their contribution to the price value is illustrated in Figure 12d.

The feature contributions obtained by PCA have been further validated through SRA and Pearson correlation (Figure 13).

Both correlations in Figure 13a and regression coefficients in Figure 13b highlight the major contribution of spatial attributes to real-estate price units. Successive to spatial attributes, the building age displays a good relationship with the price feature. The obtained SRA results are in accordance with first, second and third principal feature components extracted using PCA, and therefore SRA validates feature selection results obtained through PCA (Figure 12).

The adopted PCA-DNN model enhances the prediction accuracy of high-dimensional real-estate price units.

6 Conclusion

Accurate real-estate price prediction is an important decision-making parameter for construction professionals. Despite the importance of the topic, the literature on construction management has not investigated the house price-prediction models for the use of construction professionals. One of the obstacles to using the existing house price predictors is the high-dimensionality of the real-world real-estate price dataset, which reduces the price-prediction accuracy of the adopted predictors. The existing literature on ML-based real-estate price prediction over the small-sized dataset has not reached a good prediction accuracy. This paper adopts PCA for dataset transformation, dimensionality reduction and localisation of the influential price features. The model performance is evaluated against benchmark DNN and SRA-DNN models. The PCA-DNN model significantly outperforms the DNN and SRA-DNN models, while exhibiting 90% accuracy with a good generalisation ability. The results suggest that the application of PCA not only reduces the high-dimensionality of the dataset but also enhances the quality of the encoded feature attributes. The observed improvement may be due to improved

DNN learning through the utilisation of the transformed numerical and categorical price feature attributes in continuous numerical format.

The proposed PCA-DNN model proved to be efficient in managing the high-dimensional real-estate price dataset. Additionally, it was found that spatial features along with building age have the most influence in the determination of the total real-estate price. The adoption of the PCA-DNN model compensates for the reduced prediction accuracy due to the limited number of price records and the categorical nature of feature columns. The model is beneficial in real-estate and construction applications, where the absence of medium and big datasets decreases the price-prediction accuracy. Furthermore, the PCA utilised in this study works over the linear dataset, and the proposed study is yet to experiment with other real-estate datasets with different feature columns such as time-varying price and inflation features. Thus, it is recommended to conduct this study using the other variations of PCA, such as Kernel and robust PCA.

Data Availability Statement

Some or all data, models or codes that support the findings of this study are available from the corresponding author upon reasonable request.

Disclosure Statement

The authors report that there are no competing interests to declare.

References

Abdul-Rahman, S., Zulkifley, N. H., Ibrahim, I., & Mutalib, S. (2021). Advanced machine learning algorithms for house price prediction: Case study in Kuala Lumpur. *International Journal of Advanced Computer Science and Applications, 12*, pp. 736-745. doi: 10.14569/IJACSA.2021.0121291.

Awad, M., & Khanna, R. (2015). Efficient Learning Machines: Theories, Concepts, and Applications for Engineers and System Designers, Efficient Learning Machines: Theories, Concepts, and Applications for Engineers and System Designers. Apress, Berkeley, CA. doi: 10.1007/978-1-4302-5990-9.

Ayesha, S., Hanif, M. K., & Talib, R. (2020). Overview and comparative study of dimensionality reduction techniques for high dimensional data. *Information Fusion, 59*, pp. 44-58. doi: 10.1016/j.inffus.2020.01.005.

Cao, Y., Ashuri, B., & Baek, M. (2018). Prediction of unit price bids of resurfacing highway projects through ensemble machine learning. *Journal of Computing in Civil Engineering, 32*, p. 04018043. doi: 10.1061/(asce)cp.1943-5487.0000788.

Chen, M., Liu, Y., Arribas-Bel, D., & Singleton, A. (2022). Assessing the value of user-generated images of urban surroundings for house price estimation. *Landscape and Urban Planning, 226*, p. 104486. doi: 10.1016/j.landurbplan.2022.104486.

Chen, J. H., Ong, C. F., Zheng, L., & Hsu, S. C. (2017). Forecasting spatial dynamics of the housing market using support vector machine. *International Journal of Strategic Property Management, 21*, pp. 273-283. doi: 10.3846/1648715X.2016.1259190.

Claesen, M., & De Moor, B. (2015). Hyperparameter Search in Machine Learning.

Gondia, A., Siam, A., El-Dakhakhni, W., & Nassar, A. H. (2020). Machine learning algorithms for construction projects delay risk prediction. *Journal of Construction Engineering and Management, 146*, p. 04019085. doi: 10.1061/(asce)co.1943-7862.0001736.

Ho, W. K. O., Tang, B.-S., & Wong, S. W. (2021). Predicting property prices with machine learning algorithms. *Journal of Property Research, 38*, pp. 48-70. doi: 10.1080/09599916.2020.1832558.

Hotelling, H. (1933). Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology, 24*, pp. 417-441. doi: 10.1037/h0071325.

Hu, L., He, S., Han, Z., Xiao, H., Su, S., Weng, M., & Cai, Z. (2019). Monitoring housing rental prices based on social media: An integrated approach of machine-learning algorithms and hedonic modeling to inform equitable housing policies. *Land Use Policy, 82*, pp. 657-673. doi: 10.1016/j.landusepol.2018.12.030.

Jiang, Z., & Shen, G. (2019). Prediction of house price based on the back propagation neural network in the Keras deep learning framework. In: *2019 6th International Conference on Systems and Informatics (ICSAI)*, pp. 1408-1412. doi: 10.1109/ICSAI48974.2019.9010071.

Khalafallah, A. (2008). Neural network based model for predicting housing market performance. *Tsinghua Science and Technology, 13*, pp. 325-328. doi: 10.1016/S1007-0214(08)70169-X.

Kim, S., & Kim, H. (2016). A new metric of absolute percentage error for intermittent demand forecasts. *International Journal of Forecasting, 32*, pp. 669-679. doi: 10.1016/j.ijforecast.2015.12.003.

Kim, H., Kwon, Y., & Choi, Y. (2020). Assessing the impact of public rental housing on the housing prices in proximity: Based on the regional and local level of price prediction models using long short-term memory (LSTM). *Sustainability, 12*, p. 7520. doi: 10.3390/su12187520.

Li, W., & Shi, H. (2011). Applying unascertained theory, principal component analysis and ACO-based artificial neural networks for real estate price determination. *Journal of Software, 6*. doi: 10.4304/jsw.6.9.1672-1679.

Luo, H., Zhao, S., & Yao, R. (2021). Determinants of housing prices in Dalian City, China: Empirical study based on hedonic price model. *Journal of Urban Planning and Development, 147*, p. 05021017. doi: 10.1061/(asce)up.1943-5444.0000698.

Pal, R. (2017). Validation methodologies. *Predictive Modeling of Drug Sensitivity,* pp. 83-107. doi: 10.1016/b978-0-12-805274-7.00004-x.

Park, B., & Kwon Bae, J. (2015). Using machine learning algorithms for housing price prediction: The case of Fairfax County,

Virginia housing data. *Expert Systems with Applications, 42*, pp. 2928-2934. doi: 10.1016/j.eswa.2014.11.040.

Patel, D. A., & Jha, K. N. (2015). Neural network model for the prediction of safe work behavior in construction projects. *Journal of Construction Engineering and Management, 141*, p. 04014066. doi: 10.1061/(asce)co.1943-7862.0000922.

Pearson, K. (1901). LIII. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science, 2*, pp. 559-572. doi: 10.1080/14786440109462720.

Peng, T.-C., & Wang, C.-C. (2022). The application of machine learning approaches on real-time apartment prices in the Tokyo metropolitan area. *Social Science Japan Journal, 25*, pp. 3-28. doi: 10.1093/ssjj/jyab029.

Phan, T. D. (2019). Housing price prediction using machine learning algorithms: The case of Melbourne City, Australia. In: *Proceedings – 2018 International Conference on Machine Learning and Data Engineering (iCMLDE)*, pp. 8-13. doi: 10.1109/iCMLDE.2018.00017.

Piao, Y., Chen, A., & Shang, Z. (2019). Housing price prediction based on CNN. In: *2019 9th International Conference on Information Science and Technology (ICIST)*. IEEE, pp. 491-495. doi: 10.1109/ICIST.2019.8836731.

Poterba, J. M. (1984). Tax subsidies to owner-occupied housing: An asset-market approach. *The Quarterly Journal of Economics, 99*, p. 729. doi: 10.2307/1883123.

Qiao, X., & Guo, H. (2014). Research on the effect of the exchange rate of RMB on housing prices based on the VAR model. In: *ICCREM 2014*. American Society of Civil Engineers, Reston, VA, pp. 1251-1259. doi: 10.1061/9780784413777.148.

Rafiei, M. H., & Adeli, H. (2016). A novel machine learning model for estimation of sale prices of real estate units. *Journal of Construction Engineering and Management, 142*, p. 04015066. doi: 10.1061/(asce)co.1943-7862.0001047.

Reddy, G. T., Reddy, M. P. K., Lakshmanna, K., Kaluri, R., Rajput, D. S., Srivastava, G., & Baker, T. (2020). Analysis of dimensionality reduction techniques on big data. *IEEE Access, 8*, pp. 54776-54788. doi: 10.1109/ACCESS.2020.2980942.

Sahibinden.com. (2021). Sahibinden Available at [www.sahibinden.com/kategori/emlak]. sahibinden.com. URL www.sahibinden.com/kategori/emlak [accessed 29 April, 2021].

Sanjar, K., Bekhzod, O., Kim, J., Paul, A., & Kim, J. (2020). Missing data imputation for geolocation-based price prediction using KNN-MCF method. *ISPRS International Journal of Geo-Information, 9*, p. 227. doi: 10.3390/ijgi9040227.

Seya, H., & Shiroi, D. (2021). A Comparison of Residential Apartment Rent Price Predictions Using a Large Data Set: Kriging Versus Deep Neural Network. Geographical Analysis 0, pp. 1-22. doi: 10.1111/gean.12283.

Shi, H. (2009). Determination of real estate price based on principal component analysis and artificial neural networks. In: *2009 2nd International Conference on Intelligent Computing Technology and Automation (ICICTA)*. IEEE, pp. 314-317. doi: 10.1109/ICICTA.2009.83.

Shiha, A., Dorra, E. M., & Nassar, K. (2020). Neural networks model for prediction of construction material prices in Egypt using macroeconomic indicators. *Journal of Construction Engineering and Management, 146*, p. 04020010. doi: 10.1061/(asce)co.1943-7862.0001785.

Son, H., Kim, C., & Kim, C. (2012). Hybrid principal component analysis and support vector machine model for predicting the cost performance of commercial building projects using pre-project planning variables. *Automation in Construction, 27*, pp. 60-66. doi: 10.1016/j.autcon.2012.05.013.

Stukhart, G. (1982). Inflation and the construction industry. *Journal of the Construction Division,* 108, pp. 546-562. doi: 10.1061/JCCEAZ.0001063.

Wang, X., & Zhang, J. (2013). Principal component analysis of influencing factors of the development of China's real estate market. In: *ICCREM 2013*. American Society of Civil Engineers, Reston, VA, pp. 1027-1035. doi: 10.1061/9780784413135.098.

Wang, F., Zou, Y., Zhang, H., & Shi, H. (2019). House price prediction approach based on deep learning and ARIMA model. In: *Proceedings of 2019 IEEE 7th International Conference on Computer Science and Network Technology (ICCSNT)*, pp. 303-307. doi: 10.1109/ICCSNT47585.2019.8962443.

Wen, H., Gui, Z., Tian, C., Song, Y., & Zhou, G. (2022). Expressway proximity effects on property prices in Hangzhou, China: Multidimensional housing submarket approach. *Journal of Urban Planning and Development, 148*, p. 04021070. doi: 10.1061/(asce)up.1943-5444.0000757.

Xiao, L., & Yan, T. (2019). Prediction of house price based on RBF neural network algorithms of principal component analysis. In: *ICIIBMS 2019 – 4th International Conference on Intelligent Informatics and Biomedical Sciences*. Institute of Electrical and Electronics Engineers Inc., pp. 315-319. doi: 10.1109/ICIIBMS46890.2019.8991474.

Yue, W., Ni, C., Tian, C., Wen, H., & Fang, L. (2020). Impacts of an urban environmental event on housing prices: Evidence from the Hangzhou Pesticide plant incident. *Journal of Urban Planning and Development, 146*, p. 04020015. doi: 10.1061/(ASCE)UP.1943-5444.0000564.

Zhai, D., Shang, Y., Wen, H., & Ye, J. (2018). Housing price, housing rent, and rent-price ratio: Evidence from 30 Cities in China. *Journal of Urban Planning and Development, 144*, p. 04017026. doi: 10.1061/(ASCE)UP.1943-5444.0000426.

Zhan, D., Kwan, M.-P., Zhang, W., Xie, C., & Zhang, J. (2021). Impact of the quality of urban settlements on housing prices in China. *Journal of Urban Planning and Development, 147*, p. 05021044. doi: 10.1061/(ASCE)UP.1943-5444.0000764.

Zhang, Q. (2021). Housing price prediction based on multiple linear regression. *Scientific Programming, 2021*, pp. 1-9. doi: 10.1155/2021/7678931.

Zhang, L., Li, T., Ma, C., & Wen, H. (2020). Measuring the spatial and temporal diffusion of urban house prices in East China. *Journal of Urban Planning and Development, 146*, p. 04020017. doi: 10.1061/(asce)up.1943-5444.0000572.

Zhang, C., Xiong, M., & Wei, X. (2022). Influence of accessibility to urban service amenities on housing prices: Evidence from Beijing. *Journal of Urban Planning and Development, 148*, p. 05021063. doi: 10.1061/(asce)up.1943-5444.0000795.

Zheng, S., & Yan, L. (2017). Influence of policy adjustment on housing prices: An empirical analysis based on Chinese data since 2008. In: *ICCREM 2016*. American Society of Civil Engineers, Reston, VA, pp. 1093-1106. doi: 10.1061/9780784480274.136.