International Journal of Data Science and Analytics (2019) 8:299–323
https://doi.org/10.1007/s41060-018-00170-0

APPLICATIONS

Location identification for real estate investment using data analytics

E. Sandeep Kumar[1] · Viswanath Talasila[1] · Naphtali Rishe[2] · T. V. Suresh Kumar[3] · S. S. Iyengar[2]

Received: 10 May 2018 / Accepted: 24 December 2018 / Published online: 14 January 2019

Abstract

The modeling and control of complex systems, such as transportation, communication, power grids or real estate, require vast amounts of data to be analyzed. The number of variables in the models of such systems is large, typically a few hundred or even thousands. Computing the relationships between these variables, extracting the dominant variables and predicting the temporal and spatial dynamics of the variables are the general focuses of data analytics research. Statistical modeling and artificial intelligence have emerged as crucial solution enablers to these problems. The problem of real estate investment involves social, governmental, environmental and financial factors. Existing work on real estate investment focuses predominantly on the trend predictions of house pricing exclusively from financial factors. In practice, real estate investment is influenced by multiple factors (stated above), and computing an optimal choice is a multivariate optimization problem and lends itself naturally to machine learning-based solutions. In this work, we focus on setting up a machine learning framework to identify an optimal location for investment, given a preference set of an investor. We consider, in this paper, the problem to only direct real estate factors (bedroom type, garage spaces, etc.), other indirect factors like social, governmental, etc., will be incorporated into future work, in the same framework. Two solution approaches are presented here: first, decision trees and principal component analysis (PCA) with K-means clustering to compute optimal locations. In the second, PCA is replaced by artificial neural networks, and both methods are contrasted. To the best of our knowledge, this is the first work where the machine learning framework is introduced to incorporate all realistic parameters influencing the real estate investment decision. The algorithms are verified on the real estate data available in the TerraFly platform.

Keywords Real estate investment · Machine learning · Artificial intelligence · Decision trees · Principal component analysis · K-means clustering · Artificial neural networks · Complex systems

✉ E. Sandeep Kumar
sandeepe31@gmail.com

Viswanath Talasila
viswanath.talasila@msrit.edu

Naphtali Rishe
rishen@cs.fiu.edu

T. V. Suresh Kumar
registrar_academic@msrit.edu

S. S. Iyengar
iyengar@cis.fiu.edu

[1] Department of Telecommunication Engineering, M.S. Ramaiah Institute of Technology, Bengaluru, Karnataka 560054, India

[2] School of Computing and Information Sciences, Florida International University, Miami Dade, FL, USA

[3] Department of Computer Applications, M.S. Ramaiah Institute of Technology, Bengaluru, Karnataka 560054, India

1 Introduction

Intelligent transportation, communication or power systems are characterized by increasingly complex heterogeneous system-level data (temporal and spatial); to this, we added user-level data, social media and other services leading to big data [1]. It has been amply demonstrated that older analytical tools are not capable of handling such data and complexity [2]. Emerging data analytic tools which are predominantly based on statistical modeling [3] and machine learning techniques [4] are the solution enablers for the modeling, analysis and control of such systems [5].

The structure of real estate investment is more complex [6,7]. Real estate data are highly heterogeneous—house prices, type of housing, house dimensions, local community (religion, class, etc.), tax laws, financial conditions, personal and family choices, market conditions, and so on. This is further compounded by environmental factors, short- and long-term temporal variations, education qualifications and what not!. A realistic investment decision often takes into account multiple factors at once [8]. Much of the current research has focused on the prediction of the real estate price, without formally focusing on computing an optimal investment location [9–14].

There are many reasons why an investor may not know the specific location for investment. A simple reason may be that an investor is new to the city. A more involved reason is that even though an investor is native to the city, it is logically impossible to narrow down to a very specific location—at best a small geographical area can be identified. However, in big cities even a small area can easily compromise thousands of dwellings and commercial property; further, even the small area is often highly heterogeneous (in terms of people, establishments, facilities, etc.). Focusing only on price trends does not address the multiple concerns of an investor [15,16].

Choosing a good location for investment is very crucial since it is dependent on a huge number of user's requirements. It may be based on job availability, economic status of people, availability of restaurants, low criminal activities and safety, public transportation facility, availability of schools and shopping malls, and many more. This plenty of attributes makes a user's decision to select a location more complex and difficult. Under the influence of these huge number of attributes, the location selection may tend toward suboptimal decisions in location choice. Hence, an intelligent way of choosing the locations is of greater need in real estate investment. This includes the selection of best attributes among that huge number and choosing selections for a user helps him/her toward smart real investment. Thus, location is a critical real estate investment decision, and it is a non-trivial computation.

Let us consider few existing works available in the literature. In [9], authors use a linear regression model to predict the house price and provide techniques to balance supply and demand of constructed house, taking Shanghai city as the case study. Similarly, authors in [10] propose a linear regression method to predict the real estate price. In [11], authors use various machine learning algorithms to predict the real estate price and conclude on the best technique. [12–14,17,18] use ANNs to predict the real estate price. In [19], authors use ANNs for hedonic house price modeling, where they try to find the relation between the house price and the attributes. Based on this relation, they try to predict the house price at various locations. Authors tested their algorithms on the real estate data of Taranto (Italy). In [20], authors use correlation regression analysis using the least squares method to predict the real estate price for monthly and yearly price variation prediction of Moscow. In [21], authors use mobile phone data to establish a relation with the socioeconomic development (using measures like per capita income and deprivation index) using regression and classification techniques. For this purpose, they rely upon the municipality data of France. This work is similar to ours in the sense; they study mobile phone data instead of real estate data. However, the techniques that they have used is completely different. Authors in [22] use big data analytics to predict and estimate the traffic patterns for smart city applications. Authors use cell phone data to model the traffic pattern of users. In a broader perspective, their work aims toward smart city applications; however, the data and techniques are different compared to our work.

It is evident that the works are carried in the perspective of real estate price prediction, and identification of locations for investment is completely missing. A detailed state-of-the-art comparison of the work presented in this paper with the existing literature is provided in Table 1.

In this work, we set up statistical modeling and machine learning-based framework,[1] which looks into multiple attributes in each major factor (real estate, financial, social, etc.), and the best locations are computed w.r.t to each factor. However, to demonstrate this, specifically in this first paper, we focus exclusively on real estate parameters and demonstrate two approaches to compute best investment locations. In future work, we will use the same framework to analyze multiple factors and compute locations for real estate investment.

We set up the following research design: among 200 real estate attributes, an optimal attribute set of 9 are chosen (unless the investor has a different choice of attributes) using Pearson's coefficient. Out of these 9 attributes, an investor assigns values to the attributes that he/she desires.[2] These 9 attributes with the investor-assigned values are passed into a two-stage optimization, which computes best locations for investment. As an initial case, Miami Beach city data is considered. The roads, streets, avenues and so on are divided into clusters (we denote streets, roads, avenues, etc., as landmarks), and each cluster has a bunch of these landmarks. A user has to make an appropriate choice of a cluster at the start.[3] Each landmark has thousands of condominiums (also called as condo or condominium complex), and each condominium has units (can be called as condo units). The designed algorithm will identify locations (condominiums) within the landmarks of the chosen cluster. A set of top attributes (found using statistical models for that cluster) is presented to the user. He/she will select the attributes in which they are interested and adjust the values for those attributes.

[1] Since machine learning is a method under the hood of data analytics, usage of machine learning framework means the same as the data analytics framework in this paper.

[2] For example, if the number of bedrooms in a property is an attribute, a user can specify the desired number of rooms.

[3] Here a user need not select a specific landmark but in turn a cluster, which is a group of landmarks.

Table 1 Existing works and state-of-the-art comparison

Citation	Methodology employed	Comparison with our work
Y. Zhang et al., "Forecasting research on real estate prices in Shanghai" [9]	Authors use several attributes that influence the variations in real estate price and design a regression model for price prediction in a location	We identify the location for real estate investment. A correlation-based algorithm is used to select top attributes influencing the real estate price using which locations are identified for a user. However, regression and price trend analytics is the future perspective of this work
W. Wei et al., "Empirical analysis on the housing price in Harbin City based on hedonic model" [10]	Proposes a hedonic modeling based on nine different attributes. In addition, authors classify the influence of these factors on the house price into a variety of levels. They used linear regression for this purpose	We identify the location for real estate investment based on the correlation of attributes with the real estate price and use an algorithmic approach to identify the most influential attributes
Byeonghwa Park et al., "Using machine learning algorithms for housing price prediction: The case of Fairfax County, Virginia housing data" [11]	Proposes a method to determine the relation between the real estate attributes with the price. Authors use various machine learning techniques like C4.5, RIPPER, Bayesian and AdaBoost (for hedonic modeling). They conclude that RIPPER outperforms others	We identify location for the user based on his/her interests, and machine learning techniques are used for this sole purpose. We construct statistical model based on the simple correlation of attributes with the real estate price, and learning techniques are used in stages for identifying location
Ping Zhang et al. , "Application of Artificial Neural Network to Predict Real Estate Investment in Qingdao" [12]	Design an hedonic model based on artificial neural networks to predict real estate price. The attributes mentioned in the paper are not based on any optimality rules. These attributes are related to price using neural networks. The accuracy of neural network is around 92%	We have used machine learning techniques for location identification. Moreover, the attributes in our work are based on statistical modeling with optimality principles
H. Shi, "Determination of Real Estate Price Based on Principal Component Analysis and Artificial Neural Networks" [13]	Authors propose a hedonic model based on principal component analysis (PCA) and artificial neural networks (ANNs). They use PCA for reducing the dimension of data and ANNs as a learning tool to find the relation between the real estate price and its dependent attributes, and use the results for house price prediction	We have used machine learning for location identification and statistical models to identify the top attributes of the landmark
Eman Ahmed et al., "House price estimation from visual and textual features" [14]	Authors use support vector machines (SVMs) and ANNs for hedonic modeling and house price prediction. They conlude that ANNs outperfrom SVM. The attribute set includes numeric and image data	In our work, we do not predict the price, but instead the locations for real estate investment. We have used statistical modeling (which is a weighted linear summation) to get the top attributes for an user to enter his/her choices, based on which the locations are suggested
Maptitude product of Caliper (USA) [23]	A database system that asks user to enter the exact location and based on the series of queries entered by the user, set of lands and house suggestions are provided to a user. This is a web-based application for real estate investment	We do not rely on an accurate and exact location sepecification from a user. Moreover, the attributes selected are optimal set based on statistical modeling and optimality rules
Pitney Bowes [24]	A big data analytics based web application which askes user to specify his exact location and interests based on which a user is given with set of optimal suggestions. The website provides detailed analytics of the attributes including the price trends at that location	Our proposed method does not ask exact location from a user; moreover, we have used machine learning techniques to identify set of locations to the user

These attributes are passed into two layers of classification to arrive at the set of locations for investment. In the first stage, we use a decision tree which identifies one landmark. (We consider a single cluster with 9 landmarks in this work.) The output of the decision tree is passed into another classification layer which uses PCA and K-means clustering for location identification in a landmark. We propose another variant of the second layer where PCA is replaced by ANNs (rest remains same) and compare the obtained results from both methods.

The dataset on which the training and validation of these techniques were done comprises 9 landmarks and 36,500 condominium complexes. The total number of condominium units considered in the analysis is 7,300,000 in which for each condominium unit there are 200 attributes. In this work, landmarks for clustering are selected at random; however, nearest landmarks were given more preference during clustering. In our proposed solution, there are two different approaches, that are compared, and it was ensured that the data considered for training and validation were sufficiently and randomly chosen. The consistency of the validation accuracy of the technique is discussed in the later sections of this paper. For method-1 (with PCA in layer-2), the obtained validation accuracy on an average of 5 iterations for attribute selection was 96.86%. Layer-1 worked on an average accuracy of 100% consistently and Layer-2 with 90.25%. The accuracy of method-2 (which is variant of method-1 by replacing PCA with ANN) was calculated only for layer-2 since the other layers remain unchanged and was found to be 55.43%. This clearly shows that method-1 outperforms method-2, which is in detail dealt in Sect. 3. The sole idea of this paper is to discuss the use of concepts from data analytics to provide a user with intelligent way of choosing locations for investment.

The authors were guided in this work by the needs of the Realtor Association of Greater Miami (RAM), which is an industrial member of the National Science Foundation's Industry-University Cooperative Research Center for Advanced Knowledge Enablement at Florida International University, Florida Atlantic University, Dubna International University (Russia), and University of Greenwich (UK). The Center is directed by a co-author of this paper, Naphtali Rishe. RAM is a major user of real estate analytics technology developed by the Center, "TerraFly for Real Estate," and RAM's twenty thousand realtor members are expected to extensively use the outcomes of the present research once these outcomes they are fully incorporated into the present online tool.

The rest of the paper is organized as follows: Section 2 discusses statistical modeling for top attribute choice with classification layers and its techniques, Sect. 3 deals with the results obtained for attribute selection and classification algorithms, with related discussions, and finally Sect. 4 concludes the paper with closing remarks.

1.1 Assumptions

The proposed work is based on two assumptions. The first assumption is that a user (investor or a realtor) may not have a desired investment location, or wishes to compare investment opportunities in a large geographical region which is composed of many landmarks. The second assumption is that when a user is presented with a very large set of attribute to choose, in general the user will make a suboptimal choice. Thus, it is better to provide a user with the reduced (optimal) set of attributes.

1.2 Dataset

The data are obtained from *TerraFly* a database [25] managed and maintained by Florida International University (FIU) in collaboration with the US Government. The database which is a big data platform is a query-based system with complete information regarding economic, social, physical and governmental factors of selected countries. For our ease of working, we have considered the Miami Beach city of Miami Dade County, Florida, USA, as a case study. The streets, roads, boulevards (which we call as landmarks in this paper), etc., are divided into clusters. The clusters are formed randomly; however, preference is given to the nearby landmarks. Every landmark contains thousands of condominium complexes (we call simply as a condominium), and each condominium contains numerous units. This hierarchy is created by the authors, and it not available in the original database that just lists the information available in a condominium whose address has to be entered by the user in the query box.

Out of many clusters of landmarks, only one cluster comprising of nine landmarks is considered for further process; however, the same method is applied for the other clusters as well. The hierarchy is shown in Fig. 1.

For our work, we have considered the real estate data (i.e., current Multiple Listing Service (MLS) data, 2017 available in downloadable formats such as .csv,.xls, .json) of condominiums at Alton Rd, Bay Rd, Collins Ave, Dade Blvd, James Ave, Lincoln Rd, Lincoln CT, Washington Ave and West Ave. The approximate count of condominiums in every landmark was obtained from the official database of Miami Beach [26], i.e., for Alton Rd-7000 condominiums, Bay Rd-7000, Collins Ave-9000, Dade Blvd-1500, James Ave-2000, Lincoln Rd-2000, Lincoln CT-2000, Washington Ave-4000 and West Ave-2000, respectively. For our analysis from every landmark, 500 condominium data were randomly picked as a training dataset and 500 out of the remaining condominiums data as a validation dataset. Hence, one training corresponds to 4500 condominiums' data (including all landmarks), and similarly, validation corresponds to 4500 condominiums, respectively. The process of training

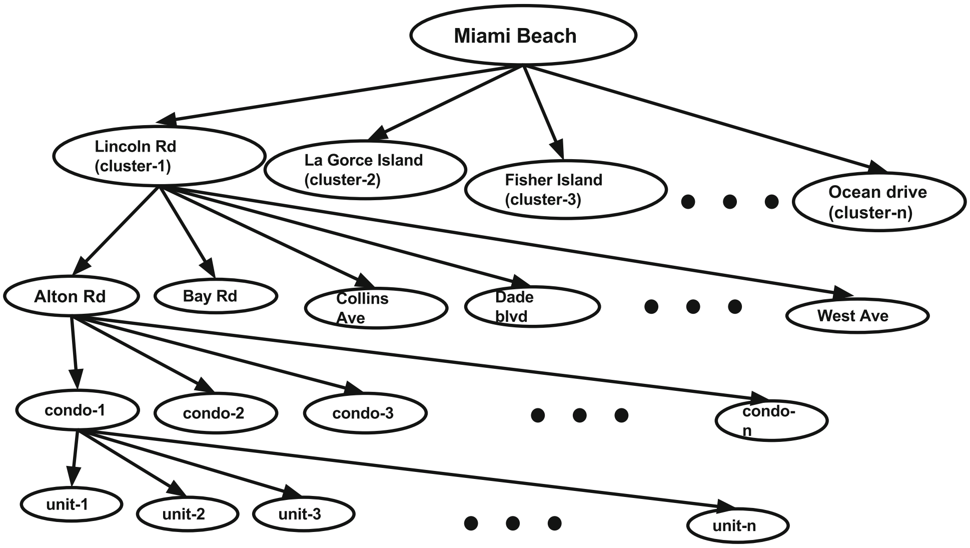


Fig. 1 Hierarchical clustering of landmarks

and validation was repeated in 5 different sets (five iterations where every time different condominium data were selected in a landmark). The results obtained from the training sets are compared with that of the validation sets, and match accuracy (validation accuracy) is noted. The process is repeated for five iteration datasets, and the average validation accuracy is quoted, which will be discussed in detail in Sect. 3.

2 Location identification using data analytics

This section discusses about the statistical modeling in detail and its associated rules used to select the top attributes within a cluster of landmarks. In addition, we will discuss the classification algorithms employed in layer-1 and layer-2 for location identification in detail.

2.1 Statistical modeling for top attributes selection

Pearson's coefficient [27,28] is used as a means to find the best attributes of real estate investment. The coefficient is found for every attribute with respect to the real estate price of that condominium in a landmark within a considered cluster. In addition, for every attribute, the normalized sample count is determined. A weighted linear summation (not a linear regression) of both these quantities determines a number (identity/label) for every condominium in a landmark; let this quantity be χ, which is shown in (1).[4] In this work, we have restricted our analysis for real estate factors (or attributes), and the rest of the factors are out of the scope of this paper.

$$\chi = (w_1 * C) + (w_2 * A) \tag{1}$$

where C is the Pearson's coefficient and A is the normalized available sample count. Let us consider an attribute, *number_of_beds* of say condominium-1 of Alton Rd. While preparing the database, there are chances that an entry might lead to NA or blank space. These data points are cleansed, and the ratio of the available data points to the total data points in that condominium is calculated.[5] Let this be A. Post-data cleansing, the correlation coefficient of that attribute with the *price per square feet* (which is real estate price) was calculated, let this be C. These two values are substituted in (1) to calculate χ value. This χ value will, in turn, determine the relation of any attribute with the *price per square feet* in that condominium. We find the χ values of all the attributes of a condominium. Based on the magnitude of χ value, we select the top attributes in a condominium. Following this, based on the frequency of occurrence (highest), we have selected top attributes of a landmark and then the top attributes of a cluster, respectively.

This is a linear constrained optimization problem defined as below:

$$\underset{C,A}{\arg\max} \quad w_1 C + w_2 A$$
$$\text{Subject to } \{-1 \leq \mathrm{C} \leq 1, 0 \leq \mathrm{A} \leq 1\} \text{ and } w_1, w_2 \in \mathbb{R}$$

[4] χ is just the representative of a condominium obtained by summation of two numbers and is not a predicted value.

[5] Here attribute linked to a condominium has data of all the units available in that condominium. Sometimes a proper entry for these units might not be available which includes NAs, incomplete words, typographical errors, and so on. These improper entries are removed, and the ratio of available data points to the total units available in that condominium is found. All the attributes associated with a condominium are available as a downloadable .csv file with condominiums units as the rows and the attributes as the columns.

The χ value embeds itself with the correlation value and the available data points information. The correlation value was chosen for the fact that it is a measure of the relation between two entities. Stronger the relation, the resulting measure is more positive which boosts the value of χ, weaker the relation the resulting measure is more negative which pulls the χ value down; if they are not related, then it has no effect on the χ value. In this work, the attribute selection algorithm focuses on the attributes that have strong relationships with the real estate price via χ.

Consider the Algorithm-1 that demonstrates the attribute selection, where w_1, w_2 are the weights as per (1), p_1 be the number of attributes selected in every landmark, q_1 be the threshold on the number of attributes selected in a cluster of landmarks, M be the top attributes of the entire landmark, M_1 be the top attributes of the entire cluster of landmarks and N be the count of number of landmarks in a cluster.

Algorithm 1: pick_attribute_cluster

```
    Begin
    Initialize: $w_1,\ w_2,\ p_1,\ q_1,\ M,\ M_1, N$
    for (iter_var in 1: number_of_condos) {
    //[6] number_of_condos was fixed as 500 since we have fixed our training and testing set consisting of 500 condominiums from a landmark, in our simulation studies
–Get the data of the condominium [iter_var] from the TerraFly database.
        for (iter_var2 in 1:number_of_attributes){
    • Read attribute[iter_var2]
    • Calculate Pearson coefficient (say C) and the normalized sample availability (say A) and find $\chi$:
          $\chi = (w_1 * C) + (w_2 * A)$        (2)
    • Save $\chi$[iter_var2]
}
–Find the top $p_1$ number of attributes based on the values of $\chi$, let this set of attributes be denoted by $z$.
$M\left[\text{iter \_var1}, p_1\right] \leftarrow z$
// $M$ stores the top attributes of all the condominiums
}
    –Pick top $p_1$ attributes from $M$ according to its frequency of occurrence. Let this set be $F$. which is the top-voted features of the landmark in a cluster.
    –Repeat this process for all the $N$ landmarks,
$M_1\left[1 : N, p_1\right] \leftarrow F$, here $M_1$ stores the top attributes of all available landmarks
    –Select $q_1$ number of attributes from $M_1$ based on the frequency of occurrence, let this set be $E$, which is the top attribute set for the entire cluster of landmarks.
End
```

2.1.1 Nonlinear summation

This section discusses about the rationale behind the choice of weighted linear summation for finding the χ value. Since χ is the identity number for a given condominium, it can also be derived from nonlinear summation. However, it consumes considerable time, which will be discussed later.

Proposition 1 *Given a landmark L with $\mathfrak{N}$ condominiums each with n attributes, then finding χ using nonlinear summation is NP complete.*

Proof Let C be the correlation of an attribute with the real estate price of a condominium and A be the normalized count of an attribute in a condominium of a landmark L; then, $\chi = (w_1 * C) + (w_2 * A)$ which is a per (1). However, in (1) it is assumed that C is independent from the influence of other attributes, but if we consider inter-attribute correlation, then

$$\chi_1 = w_1 C_1 \sum_{i=1}^{n} Z_{1i} + w_2 A_1, \tag{3}$$

which is for condominium-1 of a landmark L. Equation (3) can be written as

$$\chi_1 = w_1 C_1 \{Z_{11} + Z_{12} + Z_{13} + Z_{14} \cdots + Z_{1n}\} + w_2 A_1, \tag{4}$$

where $Z_{11} = w_1 C_{11} + w_2 A_{11}$, $Z_{12} = w_1 C_{12} + w_2 A_{12}$, and so on. Similarly for condominium-2 and condominium-3, we get

$$\chi_2 = w_1 C_2 \sum_{i=1}^{n} Z_{2i} + w_2 A_2 \tag{5}$$

$$\chi_3 = w_1 C_3 \sum_{i=1}^{n} Z_{3i} + w_2 A_3 \tag{6}$$

in general for condominium-$\mathfrak{N}$, we can write

$$\chi_{\mathfrak{N}} = w_1 C_{\mathfrak{N}} \sum_{i=1}^{n} Z_{\mathfrak{N}i} + w_2 A_{\mathfrak{N}} \tag{7}$$

Equation (7) can be written as

$$\chi_{\mathfrak{N}} = w_1 C_{\mathfrak{N}} \sum_{i=1}^{n} \{w_1 c_{\mathfrak{N}i} + w_2 A_{\mathfrak{N}i}\} + w_2 A_{\mathfrak{N}}, \tag{8}$$

where $\mathfrak{N} = \{1, 2, 3 \ldots\}$ in a single landmark L. Equation (8) is a nonlinear summation for χ calculation. □

[6] // represents a comment.

(i) Finding χ for T number of landmarks in a cluster is NP complete.

Let a single condominium complex have p number of units,

Correlation calculation time complexity is *O(p)* and χ calculation needs $O(p) + O(np)$ time units.

For $\mathfrak{N}$ number of condominiums in a given landmark, we have: $O(p\mathfrak{N}) + O(np\mathfrak{N})$

For T number of landmarks in a cluster: $O(p\mathfrak{N}T) + O(np\mathfrak{N}T)$ time units.

We can find χ for a cluster of landmarks in a finite time.

(ii) Reduction of a given problem

Let us consider an algorithm **ALG** that inputs condominiums in a cluster of landmarks; then,

- Algorithm **ALG** returns *YES* if it can calculate the χ values successfully.
- Returns NO if it cannot calculate χ values, which happens when the variance in an attribute of a condominium unit is zero.

  Hence, from (i) and (ii) the given problem is NP complete.

  Both linear summation and nonlinear summation of C and A result in successful χ values which are used later for classification. However, nonlinear summation consumes considerable time, and hence, we have opted weighted linear summation for further steps.

Remark 1 Given a cluster of N landmarks, top attribute set E is selected for further stages of classification.

A cluster has N number of landmarks (say Lincoln Rd cluster has Alton Rd, West Ave, Collins Ave and so on). Every landmark has thousands of condominiums. Every condominium has hundreds of units, and every unit has a set of attributes with magnitudes (say number of bedrooms, number of garage spaces and so on); a hierarchical representation is shown in Fig. 1.

First, we find the p_1 top attributes for every condominium which is set z. Later, we pick p_1 top features from the entire condominium set of a landmark; this will be set F. (We have N number of such F sets.) From N sets, we obtain E, which are the top attribute set for the entire cluster of landmarks. In the proposed research work, p_1 (number of attributes) was fixed as 10 and q_1 was fixed as 9. The attributes were selected based on (1). □

In Eq. (1), w_1 and w_2 are the weights assigned for C and A, respectively. Here, A was considered because the correlation of the attribute holds true only if there are enough data points in the considered condominium of a landmark.

The reason for selecting p_1 number of attributes (i.e., fixing threshold on the number of attributes) from the available attribute set was due to the less variance among their χ as shown in Fig. 2. The χ values of all the attributes are calculated within a condominium, and the variance among them is plotted (which is a single number). We have variance along y-axis and condominium complex ID numbers as X-axis. Five hundred condominiums were selected from every landmark, and the variance was calculated. Every dot in the plot represents a variance value (variance of χ values) of a condominium of a landmark. In the plot, it is clear that the variance of χ values in every condominium is almost between 0.05 and 0.15, which is very less. This trend repeats in all the condominiums of a landmark. In that case, all the attributes are significant in a condominium, and all must be considered for the next level (for classification stage). But to avoid computational complexity, we have fixed a threshold p_1 as 10 and q_1 as 9. Thus, we have selected 10 attributes from every condominium in a landmark, and from every landmark, we select 10 attributes and a final attribute set from a cluster of landmarks has 9 top attributes which are our set E.

According to Algorithm-1, by considering the dataset as mentioned in Sect. 1.2, the following attributes were obtained as the top attributes,

- ***Number of beds:*** Number of bedrooms available in the unit of a condominium building.
- ***Number of full baths:*** Number of full bathrooms (tub, shower, sink and toilet) available in the unit.
- ***Living area in sq. ft.:*** The space of the property where people are living.
- ***Number of garage spaces:*** Number of spaces available for parking vehicles.
- ***List price:*** Selling price of the property (land+assets) to the public.
- ***Application fee:*** Fee paid for owner's associations
- ***Year Built:*** Year in which the condominium/apartment complex built.
- ***Family Limited Property Total value 1:*** The property value accounted for taxation after all exemptions. This is for the district that does not contain schools and other facilities.
- ***Tax amount:*** The amount paid as tax for the property every year.

The obtained top attributes are the inputs (or as features) to the next consecutive layers of classification for location identification.

2.2 Multilayer classification model

In this section, we will discuss in detail about the layered approach used in identifying locations for real estate investment. We will first discuss the possible rationale for choosing multilayered classification approach. Let us consider the $Number\_of\_beds$ attribute of all the condominiums available in all the landmarks as a case study. Hypothesis

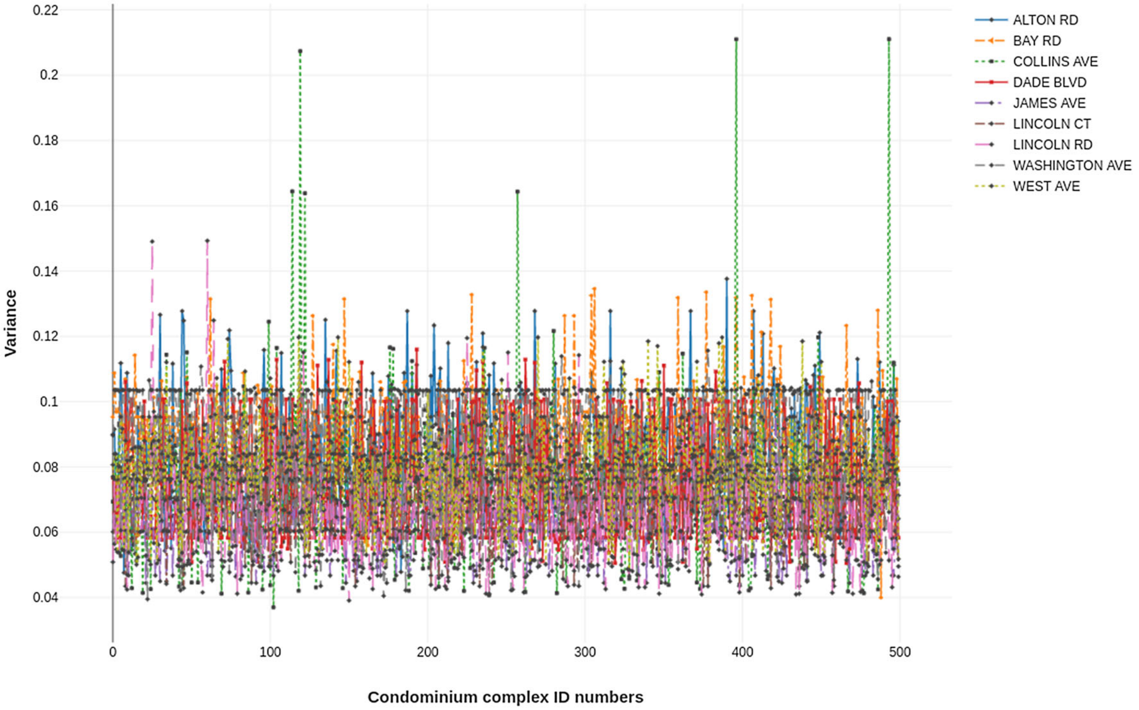


Fig. 2 Plot of variance

Table 2 Results obtained from Kolmogorov–Smirnov (K–S) test: *D* values

Landmark	Poisson	Uniform	Binomial
Alton Rd	0.0946	0.4749	0.6125
Collins Ave	0.0682	0.6246	0.7570
Bay Rd	0.099	0.5406	0.711
Lincoln Rd	0.111	0.7228	0.8685
Lincoln CT	0.1139	0.7036	0.8517
West Ave	0.1098	0.7115	0.8609
Washington Ave	0.1057	0.4781	0.8958
James Ave	0.0778	0.5432	0.4808
Dade blvd	0.0867	0.6532	0.798

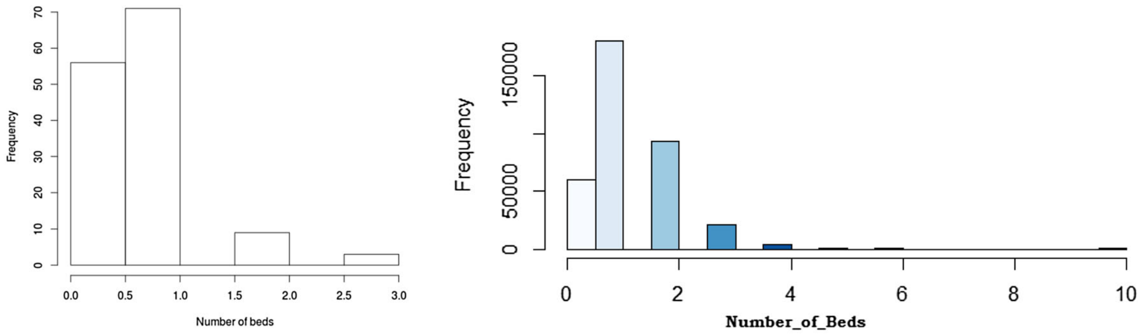
tests (also called the goodness-of-fit tests) like Kolmogorov–Smirnov (K–S) test [29] are applied to the data. These tests tell us about the probability distribution of the data (maximum likelihood from which the data are generated). From K–S test, we observe that the *D* value (the difference between the actual and assumed distributions, which serves as a conclusive parameter on the data distribution in this test) was less for Poisson distribution compared to other distributions, which is the first column in Table 2.

Also, we can see the distribution in the histogram plot of Fig. 3, where the shape of the plot qualitatively concludes that it is a Poisson distribution. The same test was performed on the few randomly chosen condominiums of the landmarks. It was still observed that the probability distribution is the same. To obtain better classification, the probability distribution of the *Number_of_beds* attribute of one landmark should not match with the other with a similar mean and variance. This results in a poor decision boundary for the classification; then, any classification technique will have poor accuracy. In our case, for the *Number_of_beds* attribute, a test was conducted to verify on three distributions, namely Poisson, uniform and binomial.[7] It was found that the data belong to the Poisson distribution with almost similar mean, in every landmark. Hence, it was decided that the identification of locations for investment is not a single layer, but a multiple-layer classification problem, where in the first layer, we used decision trees that identify landmarks, and in the second layer, principal component analysis (PCA) and K-means clustering to identify set of condominiums (we call locations) in that landmark that match user's interest.

2.2.1 Decision tree for layer-1 classification

In this section, we will deal with the construction of decision trees and its related aspects. The decision tree in our work follows the working principle of ID3-algorithm [30]. The leaf node of this tree is the landmark, and the rest

[7] We have restricted our work for these three distributions of discrete class; rest will be considered in our future work. It is intuitive that the data do not belong to geometric distribution.



(a) 100^{th} condominium of Lincoln Rd, Miami Beach

(b) All condominiums of Lincoln Rd, Miami Beach

Fig. 3 Histogram plot of *Number_of _beds* attribute

of the nodes are the attributes that are obtained according to *Algorithm-1*. The constructed decision tree is shown in Fig. 4. The attributes (set E according to Algorithm-1) are entered by the user with suitable magnitudes. This option entry of a user is converted into a string of 1's and 0's. Presently, we neglect the magnitudes (which will be used in layer-2 classification and discussed later in detail in this section). This means that we extract the information about whether a user is interested in this attribute or not, which results in a binary string. Consider an example, suppose a user is interested in *number of beds* and *number of garage spaces*; then, the tree traversal is shown in Fig. 5.

An attribute is selected as the *root node* of a tree based on the information gain of that attribute. The attribute with the highest information gain is the root and followed by that, the attributes occupy the next levels according to their decreasing order of information gain.

For this purpose, we decide the leaf nodes of the tree first, and arbitrarily the nodes are placed at the different levels including root. Later, the nodes are reshuffled based on the information content of the nodes (according to ID3) to obtain a final trained decision tree. In this direction, every tree has one or more nodes with high information content. If it is a single attribute, that itself becomes the root node; if there are more than one contenders with the same information content, for the root position, the tie is broken arbitrarily and one among them is placed at the root.

The landmark prediction from the designed tree uses a method called *highest magnitude win approach*. Recall that the user's option entry was converted into a vector and each binary bit in that vector is a *yes* or a *no* decision in a tree. In addition, we have E set, which comprises top attributes of the landmark cluster. Consider a specific case, without loss of generality, a user is interested in say, *number of beds*, *number of garage spaces* and *number of full baths* among the top attributes discussed earlier; then, the vector is 1101 0000 0 (as per the order of attributes mentioned in Sect. 2).

The set of E attributes has an associated χ value, that is obtained by averaging χ values of all the condominiums in that landmark. Therefore, every landmark has set of χ values associated with this E attribute set. Suppose a user has entered *number of beds*, then the corresponding χ values of all the landmarks are compared and the landmark with the highest χ value will be considered. Together with *number of beds*, suppose now a user has entered *number of garage spaces*, then the same process was repeated and landmark with the highest χ value is selected. This process is repeated for all the entries that a user has made, and finally, we have set of landmarks, entered attributes and the χ values out of which a landmark is selected based on whichever landmark secured highest χ value compared to all the other landmarks. This landmark is tabulated in the output column (leaf node) for that specific entry of the table (for that row vector of binary bits, or a specific tree traversal case). This process is called *highest magnitude win approach*; using this approach, we decide the leaf nodes of the decision tree.

The next step is to reshuffle the attributes, and based on the leaf nodes, the root node is selected so that a decision tree always traverses in the path of highest information gain to the leaf node (landmark). The designed truth table is shown in Table 3. The binary entries in the table are all the possible combinations of user interests or the tree traversal cases. Taking the target column (in column-4) as the parent node, and considering each attribute (column-1 to column-3) at a time, we calculate an attribute information gain. Depending on the magnitude of information gain, we decide the position of that attribute in a decision tree.

After knowing the possible inputs (attributes) and outputs from a decision tree, we proceed to the structural design of the tree. Let us consider a single attribute and solve for different cases: (i) $p_{\mathrm{t}} > p_{\mathrm{f}}$, (ii) $p_{\mathrm{t}} < p_{\mathrm{f}}$, (iii) $p_{\mathrm{t}} = p_{\mathrm{f}}$, (iv)

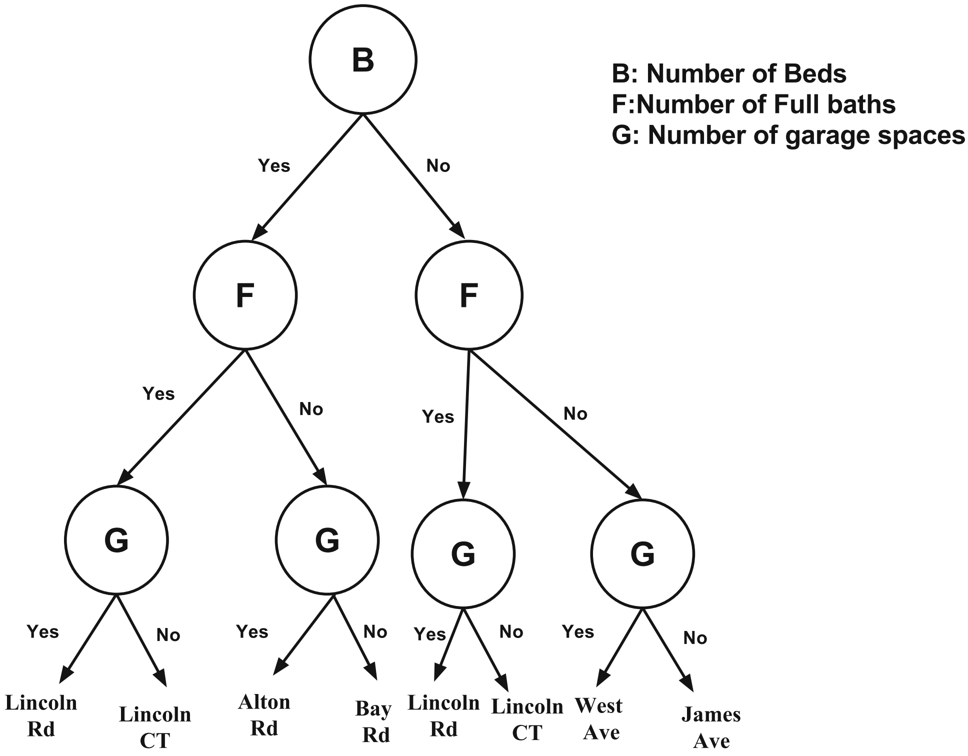


Fig. 4 Decision tree for landmark selection

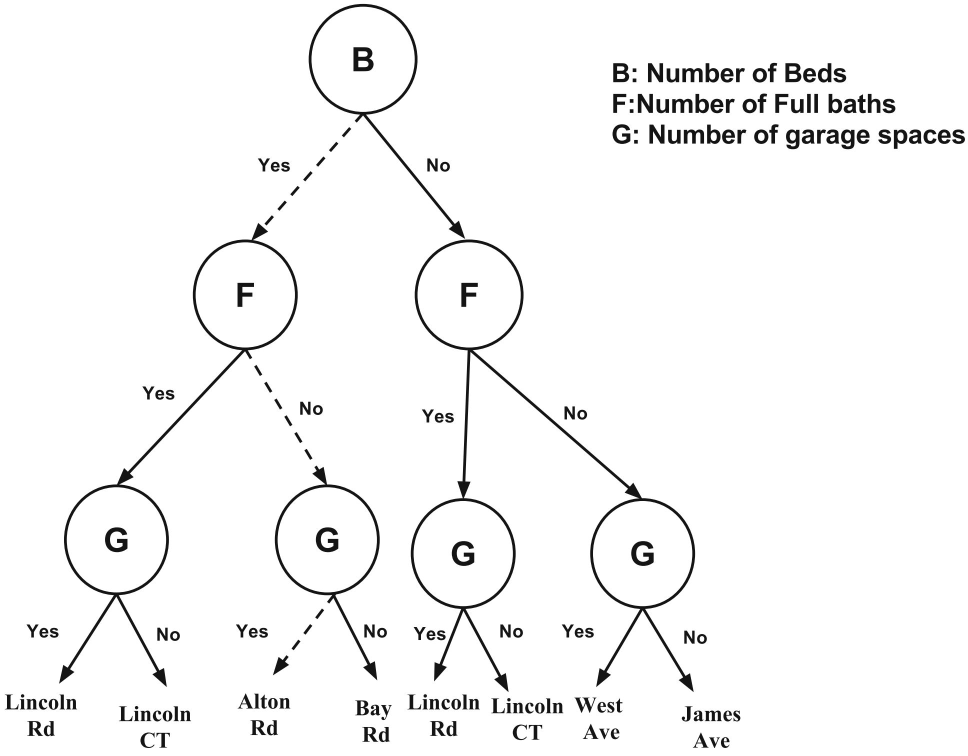


Fig. 5 Decision tree with a specific path selected

$p_t = 0$, (v) $p_f = 0$, where p_t and p_f are the probability of truths and falses in an attribute, respectively. We shall see under what conditions, the target–attribute relation gives more information gain. In addition, for every case there is no change in the probability of occurrences of instances in the target (meaning, instances occurring in a target are fixed). We show that there is one *case* among the above-mentioned five cases where the information gain is maximum for an attribute and hence a root node of that tree.

Table 3 Truth table for decision tree

Number of beds	Number of full baths	Number of garage spaces	Output
0	0	0	James Ave
0	0	1	West Ave
0	1	0	Lincoin CT
0	1	1	Lincoin Rd
1	0	0	Bay Rd
1	0	1	Alton Rd
1	1	0	Lincoin CT
1	1	1	Lincoin Rd

Procedure 1 Let $\mathscr{F} = \{f_1, f_2, f_3 \dots f_n\}$ be the set of features (attributes) $\forall \mathscr{F} \in \mathbb{R}^n$, A feature f_* is called a root of D, if the information gain $\mathrm{IG}|_{f_*} = \sup(\mathrm{IG}|_{f_*}, f_j \in \mathscr{F}$.

- We find the information gain of an attribute node obtained w.r.t parent node (target node) before and after splitting into children nodes (into attribute nodes).[8] Finding the difference between information gain before and after split w.r.t a parent determines the information gain for that attribute.
- When a parent node splits into its children nodes, eventually, the information also splits among the children. In our case, there is one child node for probability of truths and another for probability of false, respectively.
- Hence, varying number of truth and false instances in an attribute results in variation of system probabilities.[9] This results in a maximum parent–child information gain pair.

The detailed steps for this procedure are available in "Appendix A".

Complexity of decision trees: The complexity of the trees is measured in terms of total number of nodes (that depends on the number of attributes) used for its construction and depth/number of levels of a tree. A tree complexity is measured in terms of time [10] or space.[11] A tree might use different traversal techniques like pre-order, post-order, in-order and level-order.[12] There is another complexity called communication complexity[13] apart from time and space. In this paper, we have considered *time* as the complexity measure of a tree.

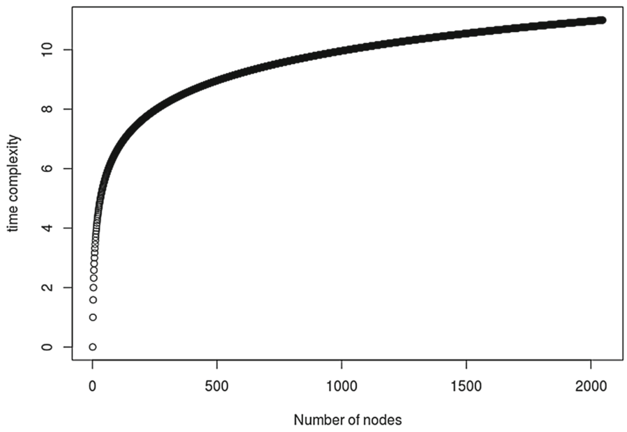


Fig. 6 Plot of variation of time complexity as a function of number of nodes

The average time complexity to traverse a binary tree is $O(\log_2 n)$, and the worst-case time complexity is $O(n)$, where n is the number of nodes in the decision tree. In our case, the time complexity is the *average time complexity* since always a part of tree is skipped during the traversals. In addition, the time complexity increases with the increase in number of nodes. In our case, with 9 features, we have 1023 nodes and the time complexity is 10. The number of nodes as a dependent variable on features is given in Eq. (9). In addition, as the number of features increases, the number of nodes in a tree increases exponentially and hence time complexity.

$$\text{nodes} = 2^{\text{features}+1} - 1 \tag{9}$$

Plot of time complexity versus number of nodes is shown in Fig. 6.

The decision tree discussed in this section is a map of a user's interest vector to the various landmarks. In fact, the truth table constructed for this binary decision considers all the possible cases of user choices. However, this tree can be modified by removing the cases that are not relevant based on the survey and opinions of the users in a geographical area. In which case, the decision tree obtained will be pruned and can reach its decision faster than the conventional tree. However,

[8] In Fig. 4, node B is split into children nodes $\{F, F\}$. In our case, in this figure it is identical, but there are nonidentical children as well, if we consider any decision tree in general. Hence, in Procedure 1, we consider a general scenario without getting into the issues of identical or nonidentical. In addition, the occurrences of 1's and 0's in Table 3 need not be always containing all the possible cases of user. It will vary according to the user's response, say no users are interested in number of full baths, then the entire column is filled with 0's.

[9] We call the probability of truths and falses of a child node, probability of landmark occurrences in the target nodes as the system probabilities, and the system being decision tree.

[10] Time complexity is the measure of the time the tree takes to arrive at a leaf node from the root node.

[11] Space complexity is the measure of the program size and the data size that occupies the memory of a system.

[12] Pre-order: the root will be processed first and then the left and right children subtrees. Post-order: the left subtree is processed first then the right subtree and finally the root. In-order: the left subtree is processed, then the root and finally the right subtree. Level-order: the processing starts from the root, then the nodes in the next level and the process continues until the traversal finishes the leaf nodes.

[13] That deals with the complexity involved in communicating nodes in a tree.

a tree constructed like in this way will always be suboptimal, since there are always chances of few important cases being neglected or over-sighted, due to the survey conducted on a limited population of users which may not generalize the entire geographical area.

There might be a case whose probability of occurrence is very minimal though, where the χ values fed to the decision trees may be same for one or more attributes, in such a case, there will be a tie between attributes and the decision tree might be unable to conclude on the landmark. Hence, in this case, manual intervention is created where the user will prioritize the attributes and choose the best attribute according to his needs. The landmark associated with that attributes will be fed to the second layer of classification.

To summarize, once a user inputs his options, the interest vector is extracted and passed into the decision tree. The tree will output the landmark (the tree in our case is the trained tree with suitable weights assigned) and hence the layer-1 classification. The accuracy of decision tree classification is discussed in Sect. 3. The next process was to identify the set of condominiums in the landmark identified by the decision tree. The condominium identification is the sole purpose of layer-2 classification which uses PCA [31] for dimension reduction and K-means algorithm [32] for clustering.

2.2.2 Principal component analysis and K-means clustering for layer 2 classification

In this section, we will discuss in detail about the second layer classification model. From Sect. 2, we have E attribute set (top attributes of a landmarks cluster); we proceed further to find principal component values and thereby principal scores. Every landmark has set of condominiums. Each condominium has set of units with its associated data (like number of bedrooms, number of garage spaces and so on). From every condominium, we select these E attributes (length p_1) and calculate principal components (which is nothing but the eigenvectors). This process reduces the dimension of the dataset into principal component vectors. We pick the first principal component since it has the maximum variance information [31]. Using PC_1, we calculate principal scores using the following equation:

$$PC\_score = \sum_{j=1}^{p_1} (\text{PC}_{1j} * attribute\_value)$$

Every unit in the condominium has its own associated magnitude. This magnitude is the *attribute_value* in the above equation and PC1 has value associated with every attribute and hence it's length is same as number of atttributes. Therefore, according to the above equation, every unit in a condominium of a landmark will have a principal score. Averaging all the principal scores gives a score for the condominium. This process was repeated for all the condominiums in a landmark. Finally, every unit in a condominium has a principal score and every condominium has a principal score in a landmark. Also, when we average the principal components (PC_1) of all the condominiums in a landmark, we get principal components for individual landmarks of a cluster.

Algorithm-2: Find the principal score of condominium and its units

```
Begin
for (condo in 1: number_of_condominiums)
{
selected _var ← condominium_data [attributes]
//attributes here is the $E$ set.
$PC_1$ ← Principal component analysis (selected_var)
$PS_x$ ← Calculate principal score of each unit in condominium,
// here $x = \{1, 2, 3 \ldots n\}$ and $n$ is length of units in a condominium.
$PS\_condo$ ← average($PS_x$)
// PS_condo is the principal score of an entire condominium.
}
End
```

We apply K-means clustering on the principal scores of condominiums in a landmark and divide it into a x number of clusters. (These clusters are different from landmark clusters discussed in Sect. 1.) Layer-2 operates on a specific landmark selected by layer-1. For this purpose, we consider the magnitude of the attributes that a user had entered (from which we extracted only the vector for decision trees), and using the principal components of that landmark, we obtain a principal score for user's entry. This score is also a representative of user's interests. This score is compared with the existing clusters of that landmark. The closest match to the centroid of principal scores is selected, and the user is concluded with the condominiums available in that cluster as the final locations for real estate investment.

2.3 Use of ANN in layer-2 instead of PCA

In this section, we discuss the variant of the method discussed in Sect. 2.2.2. Neural networks [33] are extensively used in real estate research, whether it is hedonic modeling for finding importance of the attributes or for the predictions [19,34–37]. Principal components embed itself with the nonlinearities of a system efficiently, and it is one of the widely used techniques to date. As seen in the previous Sect. 2.2.2, principal components provide a kind of ranking to the attributes that are used to find the principal scores which help in the classification process. However, in that direction ANNs can be used as an alternative to PCA since the weights gained by the attributes at the end of complete training of the network can be used for ranking the attributes as well. This ranking is obtained by using Olden method [38]. However,

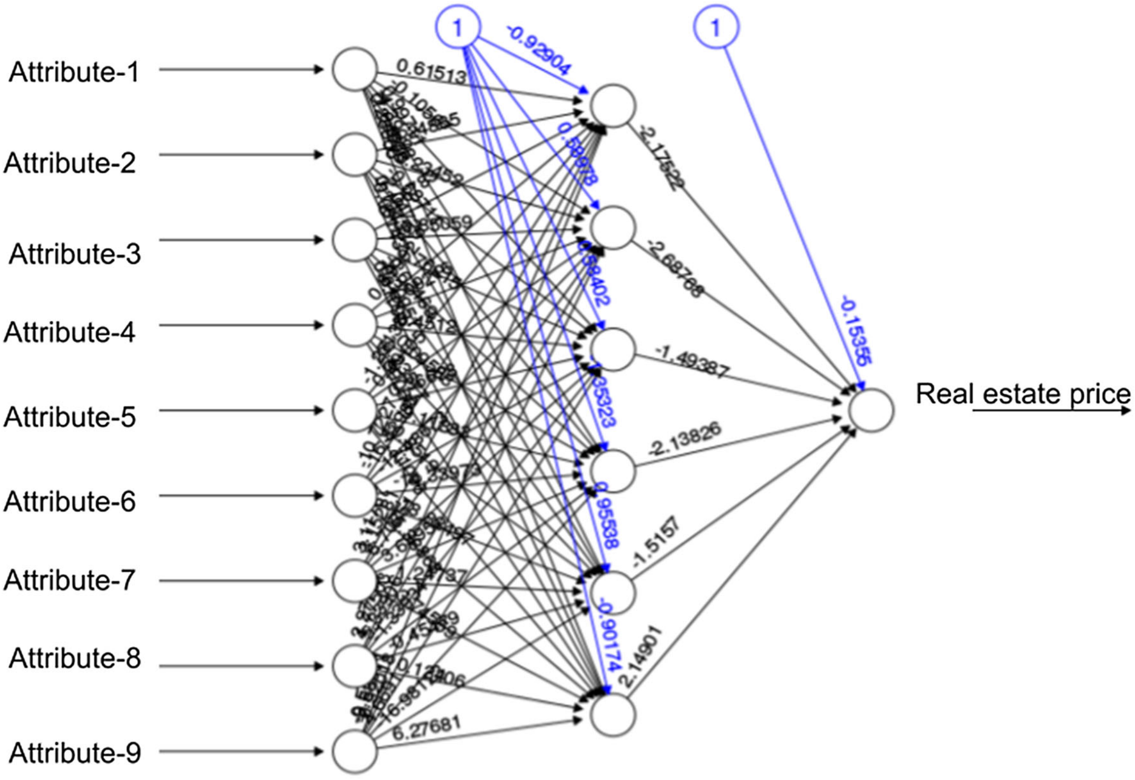


Fig. 7 Neural network architecture

to fit polynomial that considers the underlying nonlinearities in the attributes is a tedious work. Neural networks provide an easy means of fitting such a nonlinear curve into the data; in that case, a multilayer neural network will perform better than a single-layer network [37]. In addition, ANNs are representatives of the class of the learning algorithms that provide a weighted relationship between the input and output. However, it is also true that ANNs can be replaced with any machine learning algorithm that suffices the need for ranking the attributes that are used for classification in the location identification problem dealt in this paper.

The decision tree of layer-1 and K-means clustering associated with layer-2 was retained; however, PCA is replaced by ANNs in the layer-2, compared to the first method. The top attribute set E of a given cluster of landmarks were fed as input to the network, with one neuron at the output to predict the house price. Two hidden layers with each layer having $\frac{2}{3}\times$ (neurons of the previous layer) neurons were used. The network was trained for the real estate price of that condominium, while the attribute values of the condominium were fed as input to the network. The process was repeated for all the condominiums in a landmark. The network was trained separately for individual landmarks. Suitable learning rates and momentum were maintained throughout the training process relying on naive back-propagation algorithm. Olden technique [38] was applied to the trained network which ranked the attributes based on the weights gained at the end of training. The obtained Olden ranks were used as weights to calculate the score (we call this Olden_score) which is obtained individually for all the condominium units in a condominium similar to that of PC_score discussed in the previous method. Averaging the Olden_score over a condominium gives Olden score for a condominium. Applying K-means clustering on the Olden_scores will group the condominiums. This process is repeated for all the landmarks in a cluster. In every landmark, five iterations are performed, and we measure the accuracy by comparing the cluster centers obtained by applying K-means clustering on the training and the validation data (using MAE). The neural network architecture is shown in Fig. 7.

3 Results and discussions

In this section, we discuss the obtained validation accuracy results. We applied *Algorithm-1* on the dataset mentioned in Sect. 2.1. Let us consider Alton Rd as an example. This landmark has nearly 7000 condominiums and related data. We pick randomly 500 condominiums, we select top 10 attributes ($p_1 = 10$, which was set z) from every condominium, and from the combined set (M) we selected 10 attributes, which was set F, that are top 10 attributes for Alton Rd. We repeat

Table 4 Accuracy of optimal attribute selection phase

Iteration	No. of mismatches	Accuracy (%)
1	1 out of 25	96
2	0 out of 24	100
3	1 out of 25	96
4	1 out of 26	96.15
5	1 out of 26	96.15
Average		96.86

this process for all the nine landmarks in the cluster, and we get $F_1, F_2, \ldots\ F_9$. From these F's, we select 9 attributes ($q_1 = 9$) for our further analysis (set E) which is listed in Sect. 2.1. However, for accuracy check, we have considered all uniquely occurring attributes in F without imposing a threshold q_1. Let us call this set at V_1.

Now apart from those 500 condominiums selected for training, we select another 500 condominiums for validation and repeat the same process; let this set be V_2. We compare set V_1 and V_2 and check number of mismatches, that defines our accuracy of Algorithm-1. We repeated the process 5 times, and the percentage validation accuracy is tabulated. The percentage accuracy obtained for 5 iterations is shown in Table 4.

Let us repeat a similar process to check the accuracy of decision trees. By this time, we know the top attribute set with their χ values with the landmark from which they earned it, using *highest magnitude win approach*. The attributes are listed in Table 5. Consider Alton Rd, randomly selected 500 condominiums in this landmark, we select only the top attributes and calculate χ values (1). Repeat this process for all the condominiums of Alton Rd. Average all χ values of the condominiums to get χ set for Alton Rd. Repeat this process for all the landmarks in the cluster. Let us tabulate it as a 9 × 9 matrix and call it T_1. This is the training phase.

Leaving the previously selected 500 condominiums, we now randomly select another 500 from every landmark and repeat the same process. Let this be T_2. We will compare highest scores and corresponding landmarks in T_1 and T_2 (highest scores is due to *highest magnitude win approach*). We repeated this process for 5 times, and the validation accuracy was tabulated. The obtained results are shown in Table 6 of "Appendix B". We can see that there are five iteration sets each having training and validation results. In those sets, the highest magnitude for every attribute is highlighted (by comparing row-wise). It was observed that the decision tree works consistently the same way in every iteration and the winning landmarks are shown in Table 5, and consistently these landmarks remain the same, leading to decision tree accuracy of 100%.

Table 5 Highest scorers of χ value from 5 iterations

Attributes	Landmarks (based on highest magnitude)	Average χ value (from 5 iterations)
Number of beds	Alton Rd	1.338
Number of full baths	Alton Rd	1.380
Year built	Lincoln CT	1.226
Application fee	James Ave	1.235
Number of garage spaces	Alton Rd	1.233
List price	James Ave	1.894
FLP total value	Washington Ave	1.291
Living area	Alton Rd	1.375
Tax amount	Bay Rd	1.164

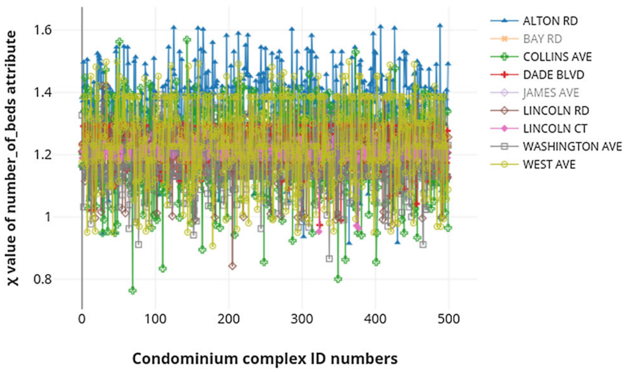


Fig. 8 Plot of χ of *Number of beds* of all landmarks

The highest scorers of χ values (that is, landmarks) are listed with their corresponding χ values. These values are in turn compared every time in the decision tree to pick a landmark based on the user's interest vector. Suppose if user is interested in *Number of beds*, *number of garage spaces* and *year built*, then their corresponding χ values are compared (1.338, 1.233, 1.226); the highest among these is 1.338 which is Alton Rd. Hence, the output of the tree will be Alton Rd. We can see in Fig. 8, where the χ values *Number of beds* attribute of all landmarks are plotted by selecting 500 condominiums in random from individual landmarks. It is clear that Alton Rd is highest compared to all the landmarks.

After deciding the landmark, the next task is to identify condominiums in that landmark, which was carried out using PCA and K-means clustering. To check the accuracy of the second layer, consider a landmark, we randomly selected 500 condominiums and calculated principal score for all the units in the condominiums and principal score for the condominium. We applied K-means clustering [32] with a need of 20 clusters in every landmark and starting seed = 30 for the clustering process. The accuracy of clustering was measured in terms of BSS/TSS ratio which is

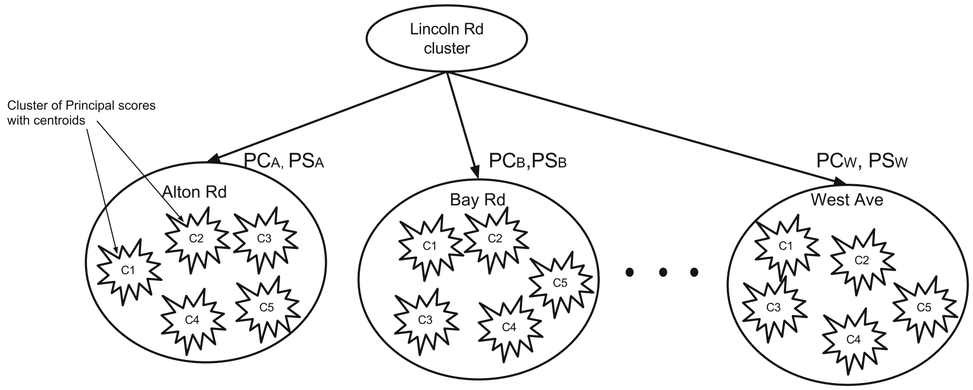


Fig. 9 Clustered condominiums in a landmark using K-means algorithm

on average 99.5% for every iteration in all the landmarks, which in turn defines goodness of clustering. In addition, finding the optimal value of K and usage of other clustering techniques instead of the K-means algorithm is an open research problem. The process of clustering is shown in Fig. 9.

Leaving the 500 condominiums selected for training, we randomly select another set of condominiums and repeat the same process. This process is the validation phase. The clusters in the training and validation are formed based on the centroids that is calculated using the K-means approach. Hence, we compare the centroids of clusters obtained by training and validation phases using mean absolute error (MAE), given by: $\mathrm{MAE} = \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)}{N}$, where N is the number of comparisons (in our case $N = 20$, since we have 20 centroid comparisons). This process was repeated for all the landmarks and for 5 such iterations. The obtained error is tabulated and shown in Table 7 (refer to "Appendix B"). It was found that the average error of the process was approximately 9.74% with correct clustering accuracy of 90.25%. For method-2, we have used a neural network with two hidden layers, one with 6 and the other with 4 neurons. The input layer had nine neurons for the attributes, output layer had one neuron for the real estate price and repetition steps (epochs) were set to 2, with learning rate 0.01, the momentum of 0.1, and the error threshold as 1e−5. Back-propagation algorithm with gradient descent was used for training. The top nine attributes are fed as the inputs, and the real estate price was taken as the output neuron. Separate neural networks are considered for a landmark. The obtained results are available in Table 8. The average accuracy in clustering of the condominiums by using ANN was 55.436%, which was observed to be less than that of using PCA with K-means clustering in layer-2. Hence, we conclude that the use of PCA gives better results than ANNs (Fig. 9).

Once we obtain the location for investment, a user might be interested to know which attribute is dominant, an effect of natural calamities on the real estate attributes and so on, in that location. Hence, we visualize the real estate scenario as a complex network system, to provide an overall picture of the real estate scenario, which is a future perspective of this paper. In addition, readers who are interested in the complete list of attributes of real estate, social and other factors are requested to obtain through TerraFly database access directly. The list scales to approximately thousand attributes including all factors.

4 Conclusions

The analysis of large-scale complex systems requires parsing through big data; machine learning and artificial intelligence have emerged as major solution enablers for these problems. In this work, we have demonstrated that real estate investment requires the analysis of hundreds of attributes in the analysis process, across thousands of investment options, and it qualifies as a large-scale complex system. When additional (indirect) factors are considered—governmental, environmental, etc., it is truly a very complex problem. In this work, we focus exclusively on the direct real estate parameters and create a framework for computing an optimal location based on the investor's choices. The same framework can be easily scaled when the indirect factors are also considered in future work.

Specifically, we have adopted the TerraFly database (of Miami Beach). We develop a two-layer constrained optimization approach to identify best locations across nine actual landmarks with 200 attributes at each condominium of a landmark. Using statistical modeling, we compute nine optimal attributes (optimal w.r.t. real estate price variation). The attributes are presented to the user (or the user can use their own attribute set), and the user gives desired values to these nine attributes. These are passed onto layers of classification, where a decision tree identifies the optimal landmark, and using PCA+K-means clustering the optimal condominium complex is computed. To compare this approach with other techniques, we replace the PCA+K-means with ANN+K-means in layer 2. The landmarks obtained from the training and validation set matched perfectly with an accuracy of 100%, which is the accuracy of the layer-1 classification technique. The obtained results from layer 2 for both the training and validation sets match with an accuracy of 90.25%. In the second variant of layer-2, the resultant accuracy was 55.43%, which proved that PCA and K-means clustering perform better than ANNs with K-means clustering.

With the growing need for smart cities, there has been a sudden necessity in the novel and intelligent approaches to solving the societal problems. In this context, the techniques addressed in this work to solve the real estate location identification are novel attempts. The work unwraps various interesting results like the probability distributions of the attributes, the correlation of the attributes with the real estate price of streets/roads, and implementing unsupervised and supervised learning models with their work accuracy comparisons, on the actual real estate data with large attributed datasets obtained from an official database. Even though the paper bounds itself for only real estate data, the same method can be extended to the other factors which make the technique scalable, and knowing the behavior of the attributes helps to build a price prediction model as well.

Thus, combining AI techniques with sophisticated statistical modeling provides an automated means of location identification. The results obtained in this work prove that the developed method is a promising technique, which could be a step toward assisting users for location identification in housing and investment of smart cities.

Appendix A

Procedure 1 Let $\mathscr{F} = \{f_1, f_2, f_3 \ldots f_n\}$ be the set of features (attributes) $\forall \mathscr{F} \in \mathbb{R}^n$, A feature f_* is called a root of D, if the information gain $\text{IG}|_{f_*} = \sup(\text{IG}|_{f_*}, f_j \in \mathscr{F}$.

Steps. Let $\mathscr{F} = \{f_1, f_2, f_3 \ldots f_n\}$ be the set of features $\forall \mathscr{F} \in \mathbb{R}^n$. Let the randomness in any variable be defined by entropy:

$$H = -p\log_2 p \tag{10}$$

where p is the probability of occurrences of instances in the column of a truth table.

Let the target be $\tau = \{p_1, p_2 \ldots p_c\}$,where c is the number of class.[14]

Let D be the decision tree $\ni D : \mathscr{F} \to \tau$; we find the root of D.

We find the information before split of a parent node (in our case the output column) by $I_{BS} = -p_1\log_2 p_1 - p_2\log_2 p_2 - \cdots - p_c\log_2 p_c =$

$$\sum_{d=1}^{c} -p_d\log_2 p_d \tag{11}$$

Consider the feature $f_i \in \mathscr{F}$ having two classes (1 or 0). The net information of the children nodes is given by

$$I_{AS} = p_{\text{t}}\left(\sum_{j=1}^{c} -p_j\log_2 p_j\right) + p_{\text{f}}\left(\sum_{k=1}^{c} -p_k\log_2 p_k\right) \tag{12}$$

Let the truth occurrences in the children (the split probability of truths) be p_{t} and that for the falses be p_{f}. Let p_j and p_k be the probabilities of the target accompanied with the truths and the falses, respectively.[15] Every instances in Eq. (12) are written according to the entropy of (10). The total information gain is obtained by subtraction of (12) from (11). Therefore, $I_{\text{g}} = I_{\text{BS}} - I_{\text{AS}}$

$$I_{\text{g}} = -\sum_{d=1}^{c} p_d\log_2 p_d + p_{\text{t}}\left(\sum_{j=1}^{c} p_j\log_2 p_j\right) + p_{\text{f}}\left(\sum_{k=1}^{c} p_k\log_2 p_k\right) \tag{13}$$

The following conditions are applied throughout the root identification process.

- $0 \le p_{\text{t}} \le 1, 0 \le p_{\text{f}} \le 1 \ni p_{\text{t}} + p_{\text{f}} = 1$
- $\left\{0 \le \sum_{j=1}^{c} p_j \le 1, 0 \le \sum_{j=1}^{c} p_k \le 1\right\} \ni \left\{\sum_{j=1}^{c} p_j + \sum_{j=1}^{c} p_k = \sum_{d=1}^{c} p_d\right\}$
- $\sum_{d=1}^{c} p_d = 1$

Let us identify the root node (with the highest information gain by induction in Eq. (12) for five cases and its variants (totally eleven in the following) discussed prior):

Case 1: When $p_{\text{t}} = 1, p_j = 0 \bigvee p_{\text{t}} = 0, p_j = p_d \; with \; p_j + p_k = p_d, p_{\text{t}} + p_{\text{f}} = 1$.

If $p_j = 0$, we have $p_k = p_d$; substituting in (13) and changing the limits, we have:

[14] In our case, there are nine landmarks; hence, $c = 9$ and $p_1, p_2 \ldots$ are the probabilities of their occurrences.

[15] Let us consider the truth table in Table 3. For the attribute *Number of Beds*, $p_{\text{t}} = \frac{4}{8} = 0.5, p_{\text{f}} = \frac{4}{8} = 0.5$. The landmarks associated with the falses are: James Ave, West Ave, Lincoln CT, Lincoln Rd; similarly, the landmarks associated with the truths are: Bay Rd, Alton Rd, Lincoln CT, Lincoln Rd. Hence, $p_j = \{\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}\}$ and same for p_k.

$$
\begin{aligned}
I_{\mathrm{g}} &= -\sum_{d=1}^{c} p_d \log_2 p_d + (1-p_{\mathrm{t}}) \sum_{d=1}^{c} p_d \log_2 p_d \\
&= -\sum_{d=1}^{c} p_d \log_2 p_d
\end{aligned} \tag{14}
$$

Case 2: When $p_{\mathrm{t}} = 1, p_j = P_d \bigvee p_{\mathrm{t}} = 0, p_j = 0$, *with* $p_j + p_k = p_d, p_{\mathrm{t}} + p_{\mathrm{f}} = 1$.

If $p_j = p_d$, we have $p_k = 0$; substituting in (13) and changing the limits we have:

$$
I_{\mathrm{g}} = -\sum_{d=1}^{c} p_d \log_2 p_d + \sum_{d=1}^{c} p_d \log_2 p_d = 0 \tag{15}
$$

Case 3: When $p_{\mathrm{t}} = 1, p_j = p_k \bigvee p_{\mathrm{t}} = 0, p_j = p_k$ with $p_j + p_k = p_d, p_{\mathrm{t}} + p_{\mathrm{f}} = 1$.

If $p_j = p_k$, we have $p_k = \frac{p_d}{2}$; substituting in (6) and on further simplification, we have:

$$
I_{\mathrm{g}} = -\sum_{d=1}^{c} p_d \log_2 p_d + \sum_{d=1}^{c} \Big(p_d \log_2 p_d - p_d \Big) = -\sum_{d=1}^{c} p_d \tag{16}
$$

Case 4: When $0 < p_{\mathrm{t}} < \frac{1}{2}, p_j = p_d$ with $p_j + p_k = p_d, p_{\mathrm{t}} + p_{\mathrm{f}} = 1$

If $p_j = p_d$, then $p_k = 0$; substituting in (13) and on further simplification, we have:

$$
\begin{aligned}
I_{\mathrm{g}} &= -\sum_{d=1}^{c} p_d \log_2 p_d + p_{\mathrm{t}} \sum_{d=1}^{c} \Big(p_d \log_2 p_d \Big) \\
&= (p_{\mathrm{t}} - 1) \sum_{d=1}^{c} p_d \log_2 p_d \succ -\frac{1}{2} \sum_{d=1}^{c} p_d \log_2 p_d
\end{aligned} \tag{17}
$$

Case 5: When $0 < p_{\mathrm{t}} < \frac{1}{2}, p_j = 0$ with $p_j + p_k = p_d, p_{\mathrm{t}} + p_{\mathrm{f}} = 1$

If $p_j = 0$, then $p_k = p_d$; substituting in (13) and on further simplification, we have:

$$
\begin{aligned}
I_{\mathrm{g}} &= -\sum_{d=1}^{c} p_d \log_2 p_d + (1-p_{\mathrm{t}}) \sum_{d=1}^{c} \Big(p_d \log_2 p_d \Big) \\
&= (-p_{\mathrm{t}}) \sum_{d=1}^{c} p_d \log_2 p_d
\end{aligned} \tag{18}
$$

Case 6: When $0 < p_{\mathrm{t}} < \frac{1}{2}, p_j = p_k$ with $p_j + p_k = p_d, p_{\mathrm{t}} + p_{\mathrm{f}} = 1$

If $p_j = p_k$, then $p_k = \frac{p_d}{2}$; substituting in (13) and on further simplification, we have:

$$
\begin{aligned}
I_{\mathrm{g}} &= -\sum_{d=1}^{c} p_d \log_2 p_d + \frac{1}{2} \sum_{d=1}^{c} p_d \log_2 p_d - \sum_{d=1}^{c} p_d \\
&= -\frac{1}{2} \sum_{d=1}^{c} p_d \log_2 p_d - \sum_{d=1}^{c} p_d \prec 0
\end{aligned} \tag{19}
$$

Case 7: When $p_{\mathrm{t}} > \frac{1}{2}, p_j = 0$ with $p_j + p_k = p_d, p_{\mathrm{t}} + p_{\mathrm{f}} = 1$

If $p_j = 0$, we have $p_k = p_d$; substituting in (13) and changing the limits, we have:

$$
\begin{aligned}
I_{\mathrm{g}} &= -\sum_{d=1}^{c} p_d \log_2 p_d + (1-p_{\mathrm{t}}) \sum_{d=1}^{c} p_d \log_2 p_d \\
&= -p_{\mathrm{t}} \sum_{d=1}^{c} p_d \log_2 p_d > -\frac{1}{2} \sum_{d=1}^{c} p_d \log_2 p_d
\end{aligned} \tag{20}
$$

Case 8: When $p_{\mathrm{t}} > \frac{1}{2}, p_j = p_d$ with $p_j + p_k = p_d, p_{\mathrm{t}} + p_{\mathrm{f}} = 1$

If $p_j = 0$, we have $p_k = 0$; substituting in (13) and changing the limits, we have:

$$
\begin{aligned}
I_{\mathrm{g}} &= -\sum_{d=1}^{c} p_d \log_2 p_d + p_{\mathrm{t}} \sum_{d=1}^{c} p_d \log_2 p_d \\
&= (1-p_{\mathrm{t}}) \sum_{d=1}^{c} p_d \log_2 p_d < -\frac{1}{2} \sum_{d=1}^{c} p_d \log_2 p_d
\end{aligned} \tag{21}
$$

Case 9: When $p_{\mathrm{t}} > \frac{1}{2}, p_j = p_k$ with $p_j + p_k = p_d, p_{\mathrm{t}} + p_{\mathrm{f}} = 1$

If $p_j = p_k$, then $p_k = \frac{p_d}{2}$; substituting in (13) and on further simplification, we have:

$$
I_{\mathrm{g}} = -\frac{1}{2} \sum_{d=1}^{c} p_d \log_2 p_d - \sum_{d=1}^{c} p_d < 0 \tag{22}
$$

Case 10: When $p_{\mathrm{t}} = \frac{1}{2}, p_j = p_d \bigvee p_{\mathrm{t}} = \frac{1}{2}, p_j = p_d$ with $p_j + p_k = p_d, p_{\mathrm{t}} + p_{\mathrm{f}} = 1$

$$
\begin{aligned}
I_{\mathrm{g}} &= -\sum_{d=1}^{c} p_d \log_2 p_d + (1-\frac{1}{2}) \sum_{d=1}^{c} \Big(p_d \log_2 p_d \Big) \\
&= -\frac{1}{2} \sum_{d=1}^{c} p_d \log_2 p_d
\end{aligned} \tag{23}
$$

Case 11: When $p_{\mathrm{t}} = \frac{1}{2}, p_j = p_k$ with $p_j + p_k = p_d, p_{\mathrm{t}} + p_{\mathrm{f}} = 1$

$$
I_{\mathrm{g}} = -\frac{1}{2} \sum_{d=1}^{c} p_d \log_2 p_d - \frac{1}{2} \sum_{d=1}^{c} p_d < 0 \tag{24}
$$

For remaining conditions, we can apply (13) to obtain information gain, which gives the maximum information gain of a tree. Let us analyze the above cases; the conditions used to obtain (14) are in contradiction to one another, i.e., $p_{\mathrm{t}} = 1$ and $p_j = 0$ cannot happen at the same time. Hence, this case can never happen in a decision tree. I_{g} in Eqs. (17) and (20) are the optimal for the information gain and best suited for the decision tree operation. In the rest of the cases, the probability conditions do not occur due to contradiction or they do not lead to maximum information gain.

Relation between information gain I_{g} and entropy H_s: (a general result)

Let us denote the overall entropy (combined entropy of parent and children) as H_s. We find a relation between H_s and I_{g}.

$$H_s = -\sum_{d=1}^{c} p_d \log_2 p_d - p_{\mathrm{t}} \sum_{j=1}^{c} p_j \log_2 p_j - p_j \sum_{k=1}^{c} p_k \log_2 p_k \tag{25}$$

When we add Eqs. (6) and (17), we get:

$$I_{\mathrm{g}} + H_s = -2 \sum_{d=1}^{c} p_d \log_2 p_d \tag{26}$$

In (26), the R.H.S is a constant because the class probabilities in the target column will not change. Hence, we can conclude that

$$I_{\mathrm{g}} + H_s = \text{constant} \tag{27}$$

This follows the notion of a straight line with a negative slope.[16]

Simulation results of Procedure-1: We simulated the equations in MATLAB 2014. The simulation parameters were as follows: number of classes=3 (nevertheless in our work, it is a 9 class problem, because cluster has 9 landmarks, for the analysis of the theorem and simulations, let us choose number of classes as 3), the probability of classes: $p_1 = 0.1$, $p_2 = 0.1$ and $p_3 = 0.8$. Let the truth occurrences in the children (the split probability of truths) be p_{t} and that for the false be p_{f}.[17] Let p_j and p_k be the probabilities of the target accompanied with the truths and the falses, respectively. The graphs are plotted for the different conditions of p_{t}—$p_{\mathrm{t}} = 0$, $p_t = 1$, $p_{\mathrm{t}} = \frac{1}{2}$, $p_{\mathrm{t}} = 0.3$, $p_{\mathrm{t}} = 0.7$. The value $p_{\mathrm{t}} = 0.3$ is a representative of the condition $0 \leq p_{\mathrm{t}} < \frac{1}{2}$, and $p_{\mathrm{t}} = 0.7$ is a representative of the condition $p_{\mathrm{t}} > \frac{1}{2}$. Since the information gain is always positive, the iteration on p_{f} will give the same outputs/results, since $p_{\mathrm{t}} + p_{\mathrm{f}} = 1$. Let the terms associated with p_{t} be p_{k1}, p_{k2}, p_{k3} and with p_{f} be p_{f1}, p_{f2}, p_{f3}. The information gain in (13) can be written as: $I_{\mathrm{g}} = -p_1 \log_2 p_1 - p_2 \log_2 p_2 - p_3 \log_2 p_3 + p_{\mathrm{t}}\{p_{k1}\log_2 p_{k1} + p_{k2}\log_2 p_{k2} + p_{k3}\log_2 p_{k3}\} + p_{\mathrm{f}}\{p_{f1}\log_2 p_{f1} + p_{f2}\log_2 p_{f2} + p_{f3}\log_2 p_{f3}\}$.

This equation can be rewritten as: $I_{\mathrm{g}} = -p_1 \log_2 p_1 - p_2 \log_2 p_2 - p_3 \log_2 p_3 + p_{\mathrm{t}}\{p_{k1}\log_2 p_{k1} + p_{k2}\log_2 p_{k2} + p_{k3}\log_2 p_{k3}\} + (1 - p_{\mathrm{t}})\{(1 - p_{k1})\log_2(p - p_{k1}) + (1 - p_{k1})\log_2(p - p_{k2}) + (1 - p_{k1})\log_2(p - p_{k2})\}$. since $p_j + p_k = p_d$ and $p_{\mathrm{t}} + p_{\mathrm{f}} = 1$.

We vary the p_{k1}, p_{k2}, p_{k3} probabilities such that $0 \leq p_{k1} \leq p_1$, $0 \leq p_{k2} \leq p_2$ and $0 \leq p_{k3} \leq p_3$. The obtained graphs are shown in Fig. 7.

In Fig. 10a, we have fixed the truth occurrences $p_{\mathrm{t}} = 0$ (meaning the feature has only false occurrences and there are no truths), and probability of class-1 occurrences is 0.1, probability of class-2 occurrences is 0.1 and that of class-3 is 0.8 in the target column. In Fig. 10a, the information gain reaches maximum when $p_k = p_d$ (meaning that all the classes of the target are associated with the truths). This is a contradiction, since there are no truth occurrences in the feature; the classes cannot associate with the truths of the children nodes. Hence, we can omit this condition and the system configuration (set of probabilities used), though the I_{g} obtained is 0.9219 which is the maximum of all the probability configurations, and if we move along x-axis, we can see 11 lobes in the information gain plot. Each main lobe has 11 sublobes, and each sublobe has 11 points which runs vertically. This is because of the possible combinations of p_1, p_2, p_3 each having 11 instances (i.e., 0 to 0.1 in steps of 0.01). Also, there is a decreasing slope between $I_{g} and H$ which goes according to Eq. (26).

In Fig. 10c, we have repeated the simulations with $p_{\mathrm{t}} = 1$ (meaning that all are truths in the considered feature column). The maximum information gain happens to be when $p_{\mathrm{t}} = 0$ with the gain value of 0.9219. This implies that the feature column has only truths, and no classes are associated with the truths. This is a contradiction, and this will not happen at the same time. Hence, the system with the probability conditions aforementioned is neglected.

In Fig. 10e, we can notice that the information gain is symmetric when $p_{\mathrm{t}} = \frac{p_d}{2}$, where the information gain reaches exactly the half of the maximum of its value. The information gain reaches to its maximum value 0.4610 that happens when $p_{\mathrm{t}} = p_d$. It can be seen that the maximum value is exactly the half of the information gain obtained according to (13). This is not a point of operation for a decision tree because the information gain goes slightly negative at its minimum point $p_{\mathrm{t}} = \frac{p_d}{2}$ or we can assume it as 0. This is because the uncertainty in the system is beyond zero, which is a contradiction in the present scenario. But we can call the point $p_{\mathrm{t}} = p_d$ as

[16] $I_{\mathrm{g}} = -H_s + \text{constant}$.

[17] The parent node splits with the truth probability of p_{t} and false probability of p_{f}.

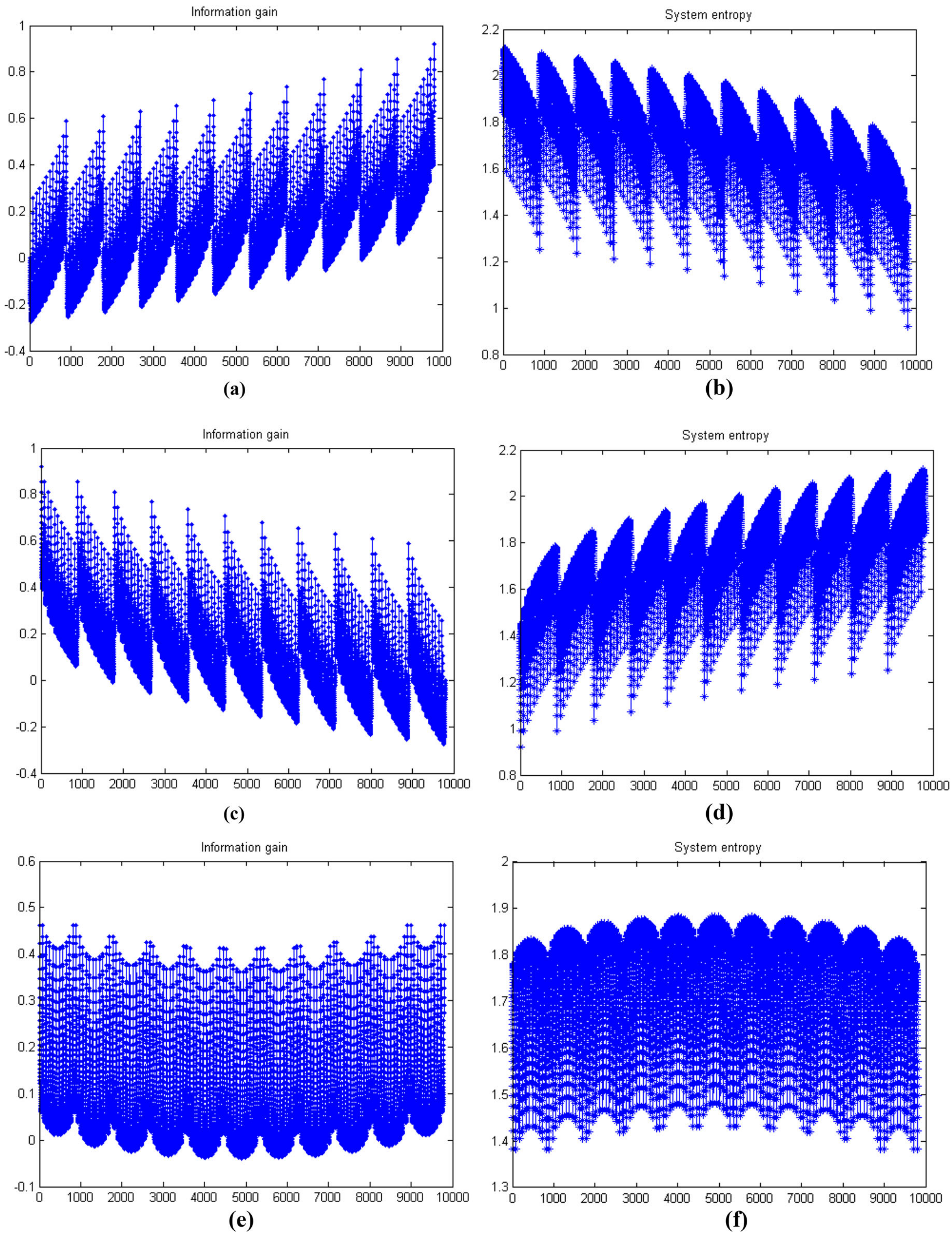


Fig. 10 **a** Plot of information gain versus possible combinations of p_t, p_k, p_j where $p_{\rm t} = 0$. **b** Plot of system entropy versus possible combinations of p_t, p_k, p_j,where $p_{\rm t} = 0$. **c** Plot of information gain versus possible combinations of p_t, p_k, p_j, where $p_{\rm t} = 1$. **d** Plot of system entropy versus possible combinations of p_t, p_k, p_j, where $p_{\rm t} = 1$. **e** Plot of information gain versus possible combinations of p_t, p_k, p_j, where $p_{\rm t} = 0.5$. **f** Plot of system entropy versus possible combinations of p_t, p_k, p_j,where $p_{\rm t} = 0.5$. **g** Plot of information gain versus possible combinations of p_t, p_k, p_j, where $p_{\rm t} = 0.3$. **h** Plot of system entropy versus possible combinations of p_t, p_k, p_j, where $p_{\rm t} = 0.3$. **i** Plot of information gain versus possible combinations of p_t, p_k, p_j, where $p_{\rm t} = 0.7$. **j** Plot of system entropy versus possible combinations of p_t, p_k, p_j, where $p_{\rm t} = 0.7$

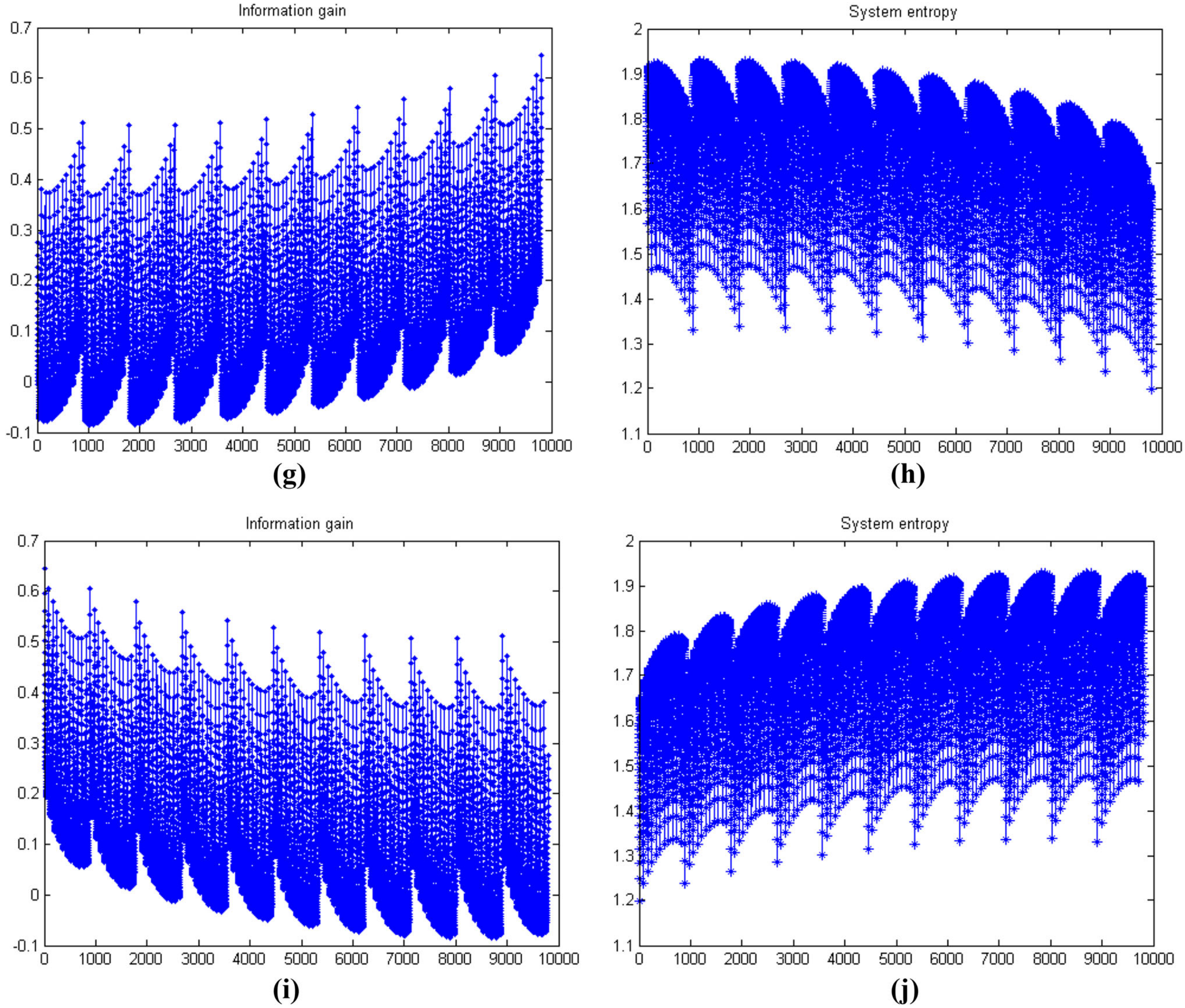


Fig. 10 continued

the equilibrium point of operation. There is no gain; neither there is loss. The information of parent gets split among the children nodes equally.

Figure 10g is the case when $p_t = 0.3$, an instance where $0 < p_t < \frac{1}{2}$; we get the maximum information gain of 0.6453 when $p_t = p_d$. It was also found that the information gain is always >0.4610, which is according to Eq. (16). It is clear that the information gain has a hard threshold where it stays always above. The feature with $0 < p_t < \frac{1}{2}$ has maximum gain when all the classes are associated with truths itself. Less truth probability with all classes associated with it gives the optimal information gain.

Figure 10i is the case when $p_t = 0.7$, an instance where $p_t > \frac{1}{2}$; we get the maximum information gain of 0.6453 when $p_t = 0$. It was also found that the information gain is always >0.4610, which is according to Eq. (20). It is clear that the information gain has a hard threshold, and it always stays above that. The feature with $p_t > \frac{1}{2}$ has maximum gain when all the classes are associated with false, meaning that none are associated with the truths. Even though the classes are associated with the false, the parent can get the maximum information gain in this case as well. We conclude that the probability conditions mentioned in *Case 4* and *Case 7* are the best conditions to choose an attribute as the root node. In other words, whichever attributes satisfies the conditions of Case 4 and Case 7 are placed as the root node of a tree.

Figure 10b, d, f, h, j are the plot of system entropy versus system probabilities, and is according to Eq. (26).

Appendix B

See Tables 6, 7 and 8.

Table 6 Validation of decision tree (layer-1 classification)

Iteration	Attributes	Alton Rd	Bay Rd	Collins Ave	Dade blvd	James Ave	Lincoln Rd	Lincoln CT	Wash. Ave	West Ave
1 (Training)	Number of beds	**1.343**	1.296	1.225	1.225	1.2019	1.209	1.2048	1.166	1.249
	Number of full baths	**1.385**	1.306	1.288	1.258	1.145	1.210	1.2045	1.273	1.274
	Year built	1.075	1.141	1.171	1.161	1.074	1.214	**1.230**	1.189	1.202
	Application fee	0.759	0.883	0.852	0.759	**1.229**	0.995	0.973	0.722	0.882
	Number of garage spaces	**1.243**	1.170	1.123	1.072	1.090	1.109	1.114	1.037	1.198
	List price	1.803	1.805	1.736	1.691	**1.894**	1.716	1.731	1.724	1.781
	FLP total value	1.282	1.278	1.243	1.099	0.990	1.095	1.173	**1.323**	1.268
	Living area	**1.379**	1.347	1.268	1.172	1.193	1.236	1.249	1.169	1.299
	Tax amount	1.094	**1.164**	0.932	0.991	0.124	0.845	0.884	1.088	0.999
1 (Validation)	Number of beds	**1.349**	1.304	1.219	1.229	1.193	1.209	1.207	1.167	1.217
	Number of full baths	**1.387**	1.314	1.278	1.265	1.135	1.209	1.205	1.277	1.271
	Year built	1.075	1.129	1.169	1.158	1.059	1.203	**1.234**	1.178	1.203
	Application fee	0.781	0.880	0.851	0.761	**1.242**	0.9623	0.9774	0.730	0.898
	Number of garage spaces	**1.243**	1.180	1.104	1.065	1.080	1.099	1.130	1.043	1.201
	List price	1.799	1.810	1.728	1.691	**1.894**	1.715	1.728	1.727	1.787
	FLP total value	1.272	1.285	1.234	1.115	0.981	1.0891	1.174	**1.323**	1.275
	Living area	**1.380**	1.350	1.272	1.169	1.181	1.232	1.251	1.176	1.298
	Tax amount	1.0885	**1.173**	0.887	1.010	0.086	0.8254	0.8901	1.086	1.001
2 (Training)	Number of beds	**1.342**	1.311	1.223	1.226	1.198	1.213	1.207	1.170	1.245
	Number of full baths	**1.386**	1.321	1.278	1.264	1.141	1.214	1.208	1.282	1.270
	Year built	1.075	1.114	1.189	1.1665	1.067	1.2181	**1.2182**	1.194	1.206
	Application fee	0.761	0.869	0.829	0.759	**1.234**	0.951	0.973	0.722	0.900
	Number of garage spaces	**1.233**	1.167	1.119	1.068	1.085	1.107	1.119	1.033	1.205
	List price	1.802	1.807	1.730	1.696	**1.894**	1.710	1.719	1.734	1.780
	FLP total value	1.282	1.285	1.235	1.118	0.986	1.103	1.156	**1.349**	1.272
	Living area	**1.379**	1.358	1.268	1.178	1.188	1.241	1.245	1.184	1.296
	Tax amount	1.095	**1.159**	0.902	1.0114	0.1100	0.860	0.837	1.121	0.987
2 (Validation)	Number of beds	**1.332**	1.312	1.216	1.226	1.203	1.204	1.2045	1.170	1.241
	Number of full baths	**1.375**	1.323	1.277	1.258	1.146	1.2095	1.204	1.279	1.267
	Year built	1.085	1.117	1.190	1.158	1.078	1.2095	**1.2424**	1.181	1.211
	Application fee	0.760	0.878	0.837	0.759	**1.230**	0.9373	0.9735	0.725	0.892
	Number of garage spaces	**1.230**	1.166	1.110	1.068	1.0920	1.100	1.124	1.0477	1.1887
	List price	1.807	1.806	1.725	1.899	**1.8954**	1.7031	1.738	1.724	1.782
	FLP total value	1.263	1.281	1.237	1.093	0.9885	1.099	1.191	**1.325**	1.265
	Living area	**1.368**	1.360	1.276	1.165	1.195	1.233	1.253	1.179	1.286
	Tax amount	1.062	**1.153**	0.882	0.980	0.1117	0.8767	0.931	1.079	0.974

Table 6 continued

Iteration	Attributes	Alton Rd	Bay Rd	Collins Ave	Dade blvd	James Ave	Lincoln Rd	Lincoln CT	Wash. Ave	West Ave
3 (Training)	Number of beds	**1.332**	1.302	1.060	1.226	1.199	1.215	1.205	1.170	1.249
	Number of full baths	**1.374**	1.312	1.113	1.258	1.140	1.216	1.207	1.276	1.268
	Year built	1.080	1.137	1.0135	1.151	1.077	1.208	**1.231**	1.180	1.210
	Application fee	0.753	0.872	0.7604	0.754	**1.245**	0.959	0.963	0.731	0.889
	Number of garage spaces	**1.225**	1.169	0.9561	1.068	1.088	1.112	1.119	1.035	1.193
	List price	1.804	1.806	1.5003	1.678	**1.896**	1.712	1.721	1.722	1.784
	FLP total value	1.259	1.286	1.0713	1.091	0.979	1.090	1.175	**1.321**	1.268
	Living area	**1.368**	1.353	1.098	1.169	1.188	1.238	1.249	1.179	1.291
	Tax amount	1.070	**1.168**	0.774	0.991	0.0615	0.820	0.905	1.078	0.976
3 (Validation)	Number of beds	**1.340**	1.308	1.204	1.223	1.194	1.209	1.207	1.172	1.255
	Number of full baths	**1.381**	1.317	1.267	1.258	1.136	1.214	1.209	1.282	1.277
	Year built	1.079	1.121	1.164	1.159	1.0677	1.205	**1.211**	1.185	1.201
	Application fee	0.785	0.871	0.851	0.764	**1.237**	0.941	0.974	0.723	0.896
	Number of garage spaces	**1.237**	1.170	1.103	1.0714	1.082	1.110	1.134	1.045	1.2114
	List price	1.802	1.806	1.723	1.694	**1.894**	1.697	1.718	1.732	1.790
	FLP total value	1.281	1.284	1.218	1.103	0.983	1.080	1.155	**1.328**	1.271
	Living area	**1.377**	1.354	1.251	1.167	1.184	1.232	1.244	1.179	1.310
	Tax amount	1.083	**1.163**	0.873	1.004	0.098	0.824	0.855	1.096	0.994
4 (Training)	Number of beds	**1.333**	1.304	0.991	1.222	1.204	1.207	1.205	1.171	1.241
	Number of full baths	**1.375**	1.314	1.039	1.254	1.150	1.208	1.207	1.279	1.268
	Year built	1.070	1.123	0.959	1.157	1.075	1.209	**1.231**	1.175	1.219
	Application fee	0.775	0.868	0.693	0.761	**1.219**	0.968	0.963	0.732	0.901
	Number of garage spaces	**1.23**	1.163	0.912	1.070	1.093	1.103	1.119	1.038	1.189
	List price	1.801	1.807	1.413	1.686	**1.894**	1.727	1.721	1.724	1.795
	FLP total value	1.264	1.286	1.013	1.089	0.996	1.090	1.175	**1.329**	1.278
	Living area	**1.370**	1.350	1.031	1.168	1.199	1.232	1.249	1.180	1.287
	Tax amount	1.082	**1.163**	0.7221	0.984	0.159	0.832	0.905	1.078	0.958
4 (Validation)	Number of beds	**1.347**	1.307	1.227	1.225	1.193	1.207	1.207	1.165	1.249
	Number of full baths	**1.389**	1.317	1.279	1.258	1.134	1.215	1.209	1.266	1.271
	Year built	1.070	1.120	1.172	1.159	1.060	1.201	**1.211**	1.176	1.192
	Application fee	0.761	0.872	0.851	0.752	**1.247**	0.935	0.974	0.727	0.902
	Number of garage spaces	**1.243**	1.167	1.113	1.072	1.079	1.113	1.134	1.041	1.203
	List price	1.803	1.806	1.741	1.687	**1.895**	1.701	1.718	1.711	1.790
	FLP total value	1.272	1.277	1.238	1.095	0.976	1.080	1.155	**1.298**	1.263
	Living area	**1.384**	1.352	1.272	1.172	1.179	1.232	1.244	1.165	1.302
	Tax amount	1.087	**1.158**	0.909	0.997	0.066	0.825	0.855	1.044	0.985

Table 6 continued

Iteration	Attributes	Alton Rd	Bay Rd	Collins Ave	Dade blvd	James Ave	Lincoln Rd	Lincoln CT	Wash. Ave	West Ave
5 (Training)	Number of beds	**1.342**	1.306	1.208	1.225	1.193	1.211	1.2100	1.170	1.251
	Number of full baths	**1.382**	1.319	1.265	1.260	1.133	1.212	1.210	1.281	1.276
	Year built	1.070	1.127	1.167	1.160	1.061	1.207	**1.220**	1.184	1.205
	Application fee	0.765	0.884	0.847	0.760	**1.248**	0.958	0.976	0.727	0.886
	Number of garage spaces	**1.237**	1.182	1.104	1.073	1.079	1.109	1.121	1.042	1.209
	List price	1.797	1.809	1.726	1.693	**1.895**	1.713	1.718	1.727	1.779
	FLP total value	1.261	1.285	1.225	1.105	0.976	1.092	1.159	**1.328**	1.272
	Living area	**1.379**	1.356	1.254	1.168	1.179	1.235	1.247	1.177	1.306
	Tax amount	1.093	**1.169**	0.882	1.003	0.067	0.840	0.859	1.094	1.005
5 (Validation)	Number of beds	**1.341**	1.304	1.203	1.224	1.202	1.208	1.2071	1.166	1.235
	Number of full baths	**1.384**	1.315	1.262	1.256	1.145	1.211	1.207	1.279	1.262
	Year built	1.079	1.133	1.151	1.158	1.073	1.203	**1.240**	1.187	1.221
	Application fee	0.764	0.878	0.855	0.758	**1.232**	0.946	0.9660	0.721	0.905
	Number of garage spaces	**1.238**	1.183	1.072	1.074	1.090	1.114	1.1325	1.208	1.195
	List price	1.804	1.809	1.715	1.682	**1.895**	1.707	1.729	1.732	1.783
	FLP total value	1.276	1.286	1.209	1.095	0.985	1.078	1.193	**1.341**	1.269
	Living area	**1.380**	1.356	1.238	1.172	1.192	1.230	1.257	1.172	1.281
	Tax amount	1.094	**1.168**	0.877	1.0006	0.109	0.825	0.942	1.103	0.978

Highest magnitude in that row is highligted (bolded) which goes analogous with the *highest magnitude win approach* technique discussed in the paper

Table 7 Validation of PCA+K-means (layer-2 classification)

Iterations	Alton Rd (%)	Bay Rd (%)	Collins Ave (%)	Dade Blvd (%)	James Ave (%)	Lincoln Rd (%)	Lincoln CT (%)	Wash. Ave (%)	West Ave (%)
1	13.11	16.88	7.90	10.60	5.87	11.33	6.46	4.69	17.04
2	11.46	12.72	14.15	11.07	1.92	7.15	6.63	6.77	8.13
3	10.12	7.79	10.05	11.37	18.30	7.42	10.73	5.10	7.37
4	11.02	7.69	1.900	7.17	10.44	12.09	11.57	5.34	15.32
5	5.215	11.60	6.99	7.79	10.31	11.56	10.04	3.95	26.15
Average error	10.18	11.33	8.20	9.60	9.37	9.91	9.09	5.17	14.80
Correct clustering	89.8	88.6	91.7	90.3	90.6	90.09	90.90	94.8	85.2

Table 8 Validation of ANNs+K-means (layer-2 classification, cluster centers match error)

Iterations	Alton Rd (%)	Bay Rd (%)	Collins Ave (%)	Dade Blvd (%)	James Ave (%)	Lincoln Rd (%)	Lincoln CT (%)	Wash. Ave (%)	West Ave (%)
1	25.052	28.552	73.403	35.735	45.926	54.815	38.163	28.284	17.994
2	37.073	34.248	50.306	40.923	33.347	52.032	58.906	31.585	28.865
3	36.226	38.170	38.657	57.77	48.141	71.089	58.0400	50.789	43.661
4	36.635	58.952	63.955	78.516	58.721	33.613	45.290	46.239	30.258
5	27.885	75.039	62.498	39.271	37.9008	37.125	31.056	43.663	32.355
Average error	32.57	46.99	57.76	50.44	44.807	49.733	46.29	40.11	30.62
Correct clustering	67.42	53.007	42.23	49.55	55.19	50.26	53.70	59.88	69.37

References

1. Chowdhury, M., Apon, A., Dey, K.: Data Analytics for Intelligent Transport Systems, 1st edn. Elsevier, New York City (2017)
2. Khan, N., Yaqoob, I., Hashem, I.A., Inayat, Z., Ali, W.K., Alam, M., Shiraz, M., Gani, A.: Big data: survey, technologies, opportunities, and challenges. Sci World J. **2014**, 712826 (2014)
3. Weihs, C., Ickstadt, K.: Data science: the impact of statistics. Int. J. Data Sci. Anal. **6**(3), 189–194 (2018). https://doi.org/10.1007/s41060-018-0102-5
4. Clarke, B., Fokoue, E., Zhang, H.H.: Principles and theory for data mining and machine learning. Springer (2009)
5. Skourletopoulos, G., et al.: Big data and cloud computing: a survey of the state-of-the art and research challenges. In: Mavromoustakis, C., Mastorakis, G., Dobre, C. (eds.) Advances in Mobile Cloud Computing and Big Data in the 5G Era. Studies in Big Data, vol. 22. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-45145-9_2
6. Carr, D.H., Lawson, J.A., Lawson, J., Schultz, J.: Mastering Real Estate Appraisal. Dearborn Real Estate Education, Wisconsin (2003)
7. Tang, D., Li, L.: Real estate investment decision-making based on analytic network process. IEEE International Conference on Business Intelligence and Financial Engineering. Beijing, pp. 544–547 (2009). https://doi.org/10.1109/BIFE.2009.128
8. Klimczak, K.: Determinants of real estate investment. Econ. Sociol. **3**(2), 58–66 (2010)
9. Zhang, Y., Liu, S., He, S., Fang, Z.: Forecasting research on real estate prices in Shanghai. In: 2009 IEEE International Conference on Grey Systems and Intelligent Services (GSIS 2009), Nanjing, pp. 625–629 (2009)
10. Wei, W., Guang-ji, T., Hong-rui, Z.: Empirical analysis on the housing price in Harbin City based on hedonic model. In: 2010 International Conference on Management Science and Engineering 17th Annual Conference Proceedings, Melbourne, VIC, pp. 1659–1664 (2010)
11. Park, B., Bae, J.K.: Using machine learning algorithms for housing price prediction: the case of Fairfax County, Virginia housing data. Expert Syst. Appl. **42**(6), 2928–2934 (2015)
12. Zhang, P., Ma, W., Zhang, T.: Application of artificial neural network to predict real estate investment in Qingdao. In: Future Communication, Computing, Control and Management, LNEE 141, pp. 213–219. Springer, Berlin (2012)
13. Shi, H.: Determination of real estate price based on principal component analysis and artificial neural networks. In: 2009 Second International Conference on Intelligent Computation Technology and Automation, Changsha, Hunan, pp. 314–317 (2009)
14. Ahmed, E., Moustafa, M.: House price estimation from visual and textual features. In: Computer Vision and Pattern Recognition. Cornell University Library. arXiv:1609.08399 (2016)
15. French, N., French, S.: Decision theory and real estate investment. J. Prop. Valuat. Invest. **15**(3), 226–232 (1997). https://doi.org/10.1108/14635789710184943
16. French, N.: Decion ecision theory and real estate investment. Manag. Decis. Econ. **22**, 399–410 (2001)
17. Li, L., Chu, K.H.: Prediction of real estate price variation based on economic parameters. In: 2017 International Conference on Applied System Innovation (ICASI), Sapporo, pp. 87–90 (2017). https://doi.org/10.1109/ICASI.2017.7988353
18. Sampathkumar, V., Helen Santhi, M., Vanjinathan, J.: Forecasting the land price using statistical and neural network software. Procedia Comput. Sci. **57**, 112–121 (2015)
19. Chiarazzoa, V., Caggiania, L., Marinellia, M., Ottomanelli, M.: A Neural Network based model for real estate price estimation considering environmental quality of property location. In: 17th Meeting of the EURO Working Group on Transportation, EWGT2014, 2–4 July 2014, Sevilla, Spain. Transportation Research Procedia, vol. 3, pp. 810–117 (2014)
20. Salnikovo, V.A., Mikheeva, M.: Models for predicting prices in the Moscow residential real estate market. Stud. Russ. Econ. Dev. **29**(1), 94–101 (2018)
21. Pappalardo, L., Vanhoof, M., Gabrielli, L., Smoreda, Z., Pedreschi, D., Giannotti, F.: An analytical framework to nowcast well-being using mobile phone data. Int. J. Data Sci. Anal. **2**(1–2), 75–92 (2016). https://doi.org/10.1007/s41060-016-0013-2
22. Tosi, D.: Cell phone big data to compute mobility scenarios for future smart cities. Int. J. Data Sci. Anal. **4**(4), 265–284 (2017). https://doi.org/10.1007/s41060-017-0061-2
23. "Maptitude"—real estate software. http://www.caliper.com/Maptitude/RealEstate/default.htm
24. Pitney bowes—real estate software. http://www.pitneybowes.com/
25. "Terrafly"—Geospatial Big Data Platform and Solutions. http://www.terrafly.com/
26. The condominium numbers' range was obtained from the website. http://www.miamidade.gov/pa/property_search.asp
27. Sheugh, L., Alizadeh, S.H.: A note on Pearson correlation coefficient as a metric of similarity in recommender system. In: 2015 AI & Robotics (IRANOPEN), Qazvin, pp. 1–6 (2015). https://doi.org/10.1109/RIOS.2015.7270736
28. Benesty, J., Chen, J., Huang, Y.: On the importance of the Pearson correlation coefficient in noise reduction. IEEE Trans. Audio Speech Lang. Process. **16**(4), 757–765 (2008)
29. Soong, T.T.: Fundamentals of Probability and Statistics for Engineers. Wiley, Hoboken (2004)
30. Schalkopff, R.J.: Intelligent Systems Principles, Paradigms, and Pragmatics. Jones and Bartlett Publishers, Burlington (2011)
31. Jolliffe, I.T.: Principal Component Analysis. Springer, Berlin (2002)
32. Wu, J.: Advances in K-means Clustering: A Data Mining Thinking. Springer, Berlin (2012)
33. da Silva, I.N., Hernane Spatti, D., Andrade Flauzino, R., Liboni, L.H.B., dos Reis Alves, S.F.: Artifical neural networks: a practical course. Springer (2017)
34. Kathmann, R.M.: Neural networks for the mass appraisal of real estate. Comput. J. Environ. Urban Syst. **17**(4), 373–384 (1993)
35. Lim, W.T., Wang, L., Wang, Y., Chang, Q.: Housing price prediction using neural networks. In: IEEE 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD), Changsha, pp. 518–522 (2016)
36. Wang, L., Chan, F.F., Wang, Y., Chang, Q.: Predicting public housing prices using delayed neural networks. In: 2016 IEEE Region 10 Conference (TENCON), Singapore, pp. 3589–3592 (2016)
37. Peterson, S., Flanagan, A.B.: Neural network hedonic pricing models in mass real estate appraisal. J. Real Estate Res. **31**(2), 147–164 (2009)
38. Olden, J.D., Jackson, D.A.: Illuminating the 'black-box': a randomization approach for understanding variable contributions in artificial neural networks. Ecol. Model. **154**, 135–150 (2002)