Analysis of Data Mining Techniques on Real Estate

Geetali Banerji, Kanak Saxena

Abstract—Data mining techniques are broadly classified into two classes (i) Statistical Techniques and (ii) Knowledge Discovery. The continuing rapid growth of on-line data and the widespread use of databases necessitate the development of techniques for extracting useful knowledge and for facilitating database access. This paper analyzes the results of multilayer perceptron with pace regression and suggests a very efficient pattern which can be proved beneficial for knowledge discovery. The analysis is done using real estate data set which contains 5821 tuples and 43 attributes and determines that in India’s scenario the demographic details of a person plays a very prominent role in identifying the investment behavior of a customer. In multilayer perceptron model, input layer is followed by two hidden layers. The first hidden layer contains 21 nodes as per various attribute weight age followed by second hidden layer which assigns re-processed weights to each of the 21 nodes. If we are discarding the demographic details then the model which is available consists of 13 Sigmoid nodes and there is a major change in error rate and correlation. We have used WEKA for analysis and found that in general multilayer perceptron (selected) is more efficient then pace regression (complete) in terms of statistical methods, but in Indian perception pace regression (complete) is more efficient than multilayer (selected).

Keywords—Multilayer Perceptron, Neural Network, Pace Regression

I. INTRODUCTION

This paper reflects the need of the investors or person in the interest of real estate era. Today, technology no doubt affects every aspect of person’s perception in living, learning, playing, working etc. Even the person having their own house(s), one can be interested to renovate it, or to buy a new for their comfort or for the investment. The Real estate can be divided into three categories, commercial, residential and agricultural. Here we consider only predicting that the investor is having power to invest or not. The real estate professionals worked and found the below three factors to be very critical for their survival.

- Must continuously change the building design or land as per the current trend to attract more users.
- Innovative ideas to construct buildings and
- Relationship to be formed among every participant in the building life cycle.

Thus to select the participants is one of the key question in the real estate. Thus, across a wide variety of fields, data are being collected and accumulated at a dramatic pace.

There is an urgent need for a new generation of computational theories and tools to assist humans in extracting useful information (knowledge) from the rapidly growing volumes of digital data. As large data sets encompass hidden trends, which convey valuable knowledge about the data set. The derived or acquired knowledge is very helpful in predicting the behavior of the user based on the data description. It can be expressed as rules or correlations highlight the associations that exist in the data [14]. For this we use the statistical analysis and knowledge discovery. With the statistical analysis we used many regression methods as linear, isotonic, pace and found that pace is more efficient among all [10]. For the machine learning process we use multiple perceptron for predicting the investment power with learning followed by the comparison among two.

The rest of the paper is organized as follows. Section II and III represents the introduction of multilayer perceptron and pace regression respectively. Section IV deals with the analysis of the real state data with was collected with 43 attributes and 5821 tuples. Section V concludes the paper with the future directions to continue the research in this direction.

II. MULTILAYER PERCEPTRON ARCHITECTURE

Neural networks are biological systems (a k a brains) that detect patterns, make predictions and learn. The artificial ones are computer programs implementing sophisticated pattern detection and machine learning algorithms on a computer to build predictive models from large historical databases. Despite the fact that scientists are still far from understanding the human brain let alone mimicking it, neural networks that run on computers can do some of the things that people can do. Because of the origins of the techniques and some of their early successes the techniques have enjoyed a great deal of interest. To understand how neural networks can detect patterns in a database an analogy is often made that they “learn” to detect these patterns and make better predictions in a similar way to the way that human beings do. Figure 1 depicts a simple neural network. In our case the network takes in values for predictors for 42 various attributes of customers and predicts whether the person will be potential enough to invest in real estate.

![Fig.1 Neural Network](Image)
All neural networks have an input layer and an output layer, but the number of hidden layers may vary. Figure 2 is a diagram of a perceptron network with two hidden layers and four total layers:

![Multilayer Perceptron Diagram](image)

**Fig. 2 Multilayer Perceptron**

When there is more than one hidden layer, the output from one hidden layer is fed into the next hidden layer and separate weights are applied to the sum going into each layer.

### A. Training Multilayer Perceptron Networks

The goal of the training process is to find the set of weight values that will cause the output from the neural network to match the actual target values as closely as possible. This can be done in following three steps:

- Train a neural network for classification problem using the data set.
- Prune the data to obtain optimized architecture in order to decrease the complexity.
- Generating the rules (Knowledge discovery and extraction)

### B. Algorithm [11]

The algorithm for Perceptron Learning is based on the back-propagation rule discussed previously. This algorithm can be coded in any programming language. In this case we are assuming the use of the sigmoid function \( f(\text{net}) \) this is because it has a simple derivative.

1. Initialise weights and threshold.
   - Set all weights and thresholds to small random values.
2. Present input and desired output
   - Present input \( \mathbf{X}_p = x_{0},x_{1},x_{2},...,x_{n} \), and target output \( \mathbf{T}_p = t_{0},t_{1},...,t_{m} \), where \( n \) is the number of input nodes and \( m \) is the number of output nodes. Set \( w_{0} \) to be \(-\phi\), the bias, and \( x_{0} \) to be always \( 1 \). For pattern association, \( \mathbf{X}_p \) and \( \mathbf{T}_p \) represent the patterns to be associated. For classification, \( \mathbf{T}_p \) is set to zero except for one element set to \( 1 \) that corresponds to the class that \( \mathbf{X}_p \) is in.
3. Calculate the actual output
   - Each layer calculates the following:
     \[ y_{pj} = f \left( \sum w_{pj}x_{j} + w_{p0} \right) \]
   - This is then passed to the next layer as an input. The final layer outputs values \( o_{pj} \).
4. Adapt weights
   - Starting from the output we now work backwards.
     \[ w_{j}(t+1) = w_{j}(t) + \eta p_{j}o_{pj} \]
   - \( \eta \) is a gain term and \( p_{j} \) is an error term for pattern \( p \) on node \( j \).
   - For output units
     \[ p_{j} = k_{j}(1 - o_{pj})(t - o_{pj}) \]
   - For hidden units

\[ p_{ij} = k_{o}(1 - o_{ij})(t - o_{ij}) \]

Where the sum (in the [brackets]) is over the \( k \) nodes in the layer above node \( j \).

### III. PACE REGRESSION

Pace regression improves on classical least square regression by evaluating the effect of each variable and using a clustering analysis to improve the statistical basis for estimating their contribution to the overall regression. Its optimality in minimizing the expected prediction loss is theoretically established when the number of free parameters is infinitely large. [9] The Pace regression overcomes the dimensionality determination problem. It outperforms existing procedures for fitting linear models. Dimensionality determination, a special case of fitting linear models, turns out to be a natural by-product. Estimating a mixing distribution is an indispensable part of pace regression. It also has more general implications for empirical modeling. Pace regression is a best technique among other regression techniques [10].

### IV. ANALYSIS

The neural network model is created by presenting it with many examples of the predictor values from records in the training set (in our case 5821) and the prediction value from those same records. By comparing the correct answer obtained from the training record and the predicted answer from the neural network it is possible to slowly change the behavior of the neural network by changing the values of the link weights.

A multiple perceptron model of real estate is shown in Figure 3. Here the training data set consists of 43 attributes and testing mode with cross validation with 10 fold. The 42 attributes are input and form 21 sigmoid Nodes, the summation of attributes multiplied with their reprocessed weights becomes input for linear node which further adds weight according to sigmoid node, which forms the final output.

![Multiple Perceptron Model](image)

**Figure 3: The normalized input values are multiplied by the link weights and added together at the output.**

The factors on which comparison is done are Correlation Coefficient, Mean absolute error; Mean squared error, Relative Absolute Error, Root relative squared error and Time taken.

### A. Comparative Study

Analysis is done on two different environments in both the techniques. In first case, we have taken all the attributes (complete) and compared the patterns, results of both the methods. In second case, the same tests are performed, after discarding demographic details of customers (selected) and then compared the various results. It is found that if we are not taking care of demographic details we are not getting very accurate results as well as the error rate is increased.

224
The following paragraphs discuss the outcomes of various comparisons.

Table 1 shows the comparison between multilayer perceptron and pace regression on all as well as selected attributes. The results are generated using WEKA 3-6-2, open source software for regression analysis and data mining. It is very much clear that correlation coefficient is very high in multilayer perceptron as compared to Pace Regression. There is a major reduction in mean absolute error, root mean square error, relative absolute error and root relative square error. It proves that multilayer perceptron far better than regression methods. It also depicts that results on selected attributes are not very accurate.

In pace regression, the following pattern is generated, which considers all the attributes in general but a single pattern takes cares of various types of tuples.

\[
PFC = -2.3609 + 0.0155 \times \text{CUSTYPE} + 0.0143 \times \text{NOH} + 0.0541 \times \text{ASH} + 0.0799 \times \text{AS} + 0.0085 \times \text{CMT} + 0.0211 \times \text{NAT} + 0.0138 \times \text{CS} + 0.0175 \times \text{SCS} + 0.102 \times \text{NR} + 0.0646 \times \text{MRD} + 0.1015 \times \text{LT} + 0.1225 \times \text{OR} + 0.0313 \times \text{SNG} + 0.203 \times \text{HWT} + 0.0222 \times \text{HLE} + 0.0231 \times \text{MLE} + 0.0162 \times \text{LLL} + 0.0183 \times \text{HS} + 0.0377 \times \text{BUS} + 0.0247 \times \text{SRS} + 0.0183 \times \text{MGMT} + 0.0361 \times \text{TL} + 0.0158 \times \text{UTL} + 0.0155 \times \text{SCA} + 0.0004 \times \text{SCB1} + 0.0293 \times \text{SCB2} - 0.0122 \times \text{SCC} + 0.1113 \times \text{SCD} + 0.1259 \times \text{RH} + 0.0533 \times \text{HO} + 0.0228 \times \text{C1} + 0.0325 \times \text{CS} + 0.2876 \times \text{NC} + 0.2644 \times \text{PL} - 0.0084 \times \text{OI} + 0.0117 \times \text{INL} + 0.0057 \times \text{INM} + 0.0242 \times \text{INMH} + 0.0447 \times \text{INH} + 0.013 \times \text{INHH} + 0.0112 \times \text{AI}
\]

Whereas in case of multilayer perceptron, 21 patterns are generated according to the weight age of various attributes in the data set proves to be very accurate and flexible. Some of the patterns generated for the sigmoid nodes, as per the weights are as follows:

**Sigmoid Node 13**

\[
PFC = -0.583 + \text{CUSTYPE} \times 1.277119249 + \text{NOH} \times 0.1047869233 + \text{SH} \times 1.556254728 + \text{AS} \times 0.81146643488 + \text{CMT} \times 0.7235987211 + \text{NAT} \times 0.236509433 + \text{CS} \times 0.811424284 + \text{CSC} \times 0.5686148686 + \text{NR} \times 0.716436701 + \text{MRD} \times 0.45365077 + \text{LT} \times 1.7582964685 + \text{OR} \times 0.429904235 + \text{SNG} \times 0.7149652544 + \text{HWT} \times 0.32104538 + \text{HWC} \times 0.23256929 + \text{HLE} \times 0.349246459 + \text{MLE} \times 0.4762989686 + \text{LLL} \times 0.784441 + \text{HS} \times 0.992203801 + \text{BUS} \times -0.679912914 + \text{SRS} \times 1.740525176 + \text{MGMT} \times 0.47712932 + \text{TL} \times 0.679549441 + \text{UTL} \times 0.673512792 + \text{SCA} \times 0.897074334 + \text{CBI} \times 0.721763457 + \text{SCB2} \times 0.923256826 + \text{SCC} \times 1.630738884 + \text{SCD} \times 0.30349656 + \text{RH} \times 0.097873626 + \text{HO} \times 1.455065 + \text{218} \times 0.434058303 + \text{C2} \times 0.23460415 + \text{NC} \times 0.398478131 + \text{PL} \times 1.09940192 + \text{OI} \times 0.425862775 + \text{INL} \times 0.954166497 + \text{NM} \times 0.251016687 + \text{INMM} \times 0.932407336 + \text{INH} \times 0.482764314 + \text{INHH} \times 0.111448519 + \text{AI} \times 0.722295018
\]

Following pattern is generated for Linear Node 0, which assigns various weights to 13 sigmoid nodes.

\[
\text{Node} = \text{Sigmoid Node 1} = 4.37894006 + \text{Sigmoid Node 2} = 5.508157156 + \text{Sigmoid Node 3} = 6.261879732 + \text{Sigmoid Node 4} = 7.53343295 + \text{Sigmoid Node 5} = 6.73971076 + \text{Sigmoid Node 6} = 3.91225698 + \text{Sigmoid Node 7} = 2.78155031 + \text{Sigmoid Node 8} = 3.12085375 + \text{Sigmoid Node 9} = 5.431623086 + \text{Sigmoid Node 10} = 6.847170019 + \text{Sigmoid Node 11} = 7.617227182 + \text{Sigmoid Node 12} = 7.01057286 + \text{Sigmoid Node 13} = 7.026772582
\]

**B. Analysis of various Graphs**

Graph 1 shows a comparison between Multilayer patterns and pace regression on all attributes. It is found that in case of pace regression it treats each attribute more or less same, but in case of multilayer each pattern gives different weight age to different
attribute according to the training set. We have taken only 11 sigmoid patterns to make the graph simple.

Graph 2, 3 and 4 shows the weight age of various attributes in case of pace regression (complete set), multilayered (on average of selected attributes) and pace regression (on selected attribute) respectively. Graph 5 and graph 6 depicts the weights assigned to various sigmoid nodes in case of selected attribute set and complete attribute set respectively. Graphs 7 to 12 depict the comparison among multilayer (selected, complete) and pace regression (selected, complete).

V CONCLUSIONS

Multilayer perceptrons are capable of generalisation, that is, they classify an unknown pattern with other known patterns that share the same distinguishing features. This means noisy or incomplete inputs will be classified because of their similarity with pure and complete inputs. Secondly they are highly fault tolerant. This characteristic is also known as “graceful degradation”. Because of its distributed nature, a neural network keeps on working even when a significant fraction of its neurons and interconnections fail. Also, relearning after damage can be relatively quick. In this paper we have tested multilayer perceptron on real estate data containing 5821 tuples and 43 attributes and found that its outperformed than pace regression which is proved best among linear regression, isotonic and least median squares with various test modes [3]. Further we have tested the same methods on selected attributes and found that demographic details of a customer matters a lot in the buying behaviour of a customer.

When compared the pace and multilayer with the complete data set and selected data set, we found that the pace gives more than +10 difference in the data set attribute (No car and policy), whereas multilayer gives (LLL, SRS and Mgt).

When compared with multilayer (Complete) and pace (selected) we found pace gives a big difference in MLE, BUS, HO, C1, NC, PL, NH where as the vice versa LLL, SRS, MGMT, TL and OI.

When compared multilayer and pace on selected dataset we found NC and PI gives very high difference where as when compared with full dataset it gives difference in HWTC, SCD, RH, NC and PL.

Such analysis gives emphasis of general thinking of any Indian Scenarios what is the impact of the data which relates with caste and religion. Generally in India the question arises on three important things in a common man life food, clothes and house. Our analysis results on impact of Pace regression the way it works as it works on the weights preprocessing where as multilayer perceptron alters the weight at the time of processing. Thus, when compared Pace complete and multilayer perceptron selected, we found the optimum analysis according to the Indian condition is LLL, SRS, MGMT, TL and OI.

REFERENCES


**Table I** Results of Multilayer Perceptron and Pace Regression on Selected / Complete Attribute Set

<table>
<thead>
<tr>
<th>Method</th>
<th>Multilayer Perceptron</th>
<th>Pace Regression</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>All Attributes</td>
<td>Selected Attributes</td>
</tr>
<tr>
<td>Correlation coefficient</td>
<td>0.9404</td>
<td>0.5032</td>
</tr>
<tr>
<td>Mean absolute error</td>
<td>0.3061</td>
<td>0.3736</td>
</tr>
<tr>
<td>Root mean squared error</td>
<td>0.3003</td>
<td>0.3308</td>
</tr>
<tr>
<td>Relative absolute error</td>
<td>32.21%</td>
<td>54.39%</td>
</tr>
<tr>
<td>Root relative squared error</td>
<td>45.62%</td>
<td>67.99%</td>
</tr>
<tr>
<td>Duration (seconds)</td>
<td>103.86</td>
<td>71.22</td>
</tr>
</tbody>
</table>

**Graph 1.** Comparison of Multilayer Patterns with Pace Regression Pattern

**Graph 2.** Pace Regression on complete attribute set

**Data Dictionary**

<table>
<thead>
<tr>
<th>#</th>
<th>Name</th>
<th>Description</th>
<th>#</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CR75/100</td>
<td>Crime 75/100</td>
<td>12</td>
<td>MMAT</td>
<td>Management</td>
</tr>
<tr>
<td>2</td>
<td>I/BH</td>
<td>Income to house ratio</td>
<td>13</td>
<td>TL</td>
<td>Taxation</td>
</tr>
<tr>
<td>3</td>
<td>A/BH</td>
<td>Age to house ratio</td>
<td>14</td>
<td>LIM</td>
<td>Loan to house ratio</td>
</tr>
<tr>
<td>4</td>
<td>AL</td>
<td>Age</td>
<td>15</td>
<td>LEM</td>
<td>Loan to employee ratio</td>
</tr>
<tr>
<td>5</td>
<td>CE/P</td>
<td>Census tract</td>
<td>16</td>
<td>LSO</td>
<td>License to operate</td>
</tr>
<tr>
<td>6</td>
<td>HS/TP</td>
<td>Higher secondary school</td>
<td>17</td>
<td>LTEM</td>
<td>Loan to employee ratio</td>
</tr>
<tr>
<td>7</td>
<td>CI</td>
<td>Census tract</td>
<td>18</td>
<td>LEM</td>
<td>Loan to employee ratio</td>
</tr>
<tr>
<td>8</td>
<td>SE/H</td>
<td>Student to house ratio</td>
<td>19</td>
<td>HOS</td>
<td>Health to hospital</td>
</tr>
<tr>
<td>9</td>
<td>NP</td>
<td>Number of people</td>
<td>20</td>
<td>MDR</td>
<td>Mortgage to house</td>
</tr>
<tr>
<td>10</td>
<td>MD</td>
<td>Mortgage to house</td>
<td>21</td>
<td>LIM</td>
<td>Loan to house ratio</td>
</tr>
<tr>
<td>22</td>
<td>G/S</td>
<td>Gender to school</td>
<td>23</td>
<td>M/S</td>
<td>Mortgage to school</td>
</tr>
<tr>
<td>24</td>
<td>S/STU</td>
<td>Student to urban</td>
<td>25</td>
<td>H/S</td>
<td>Health to school</td>
</tr>
<tr>
<td>26</td>
<td>H/SSTU</td>
<td>High school to urban</td>
<td>27</td>
<td>M/S</td>
<td>Mortgage to school</td>
</tr>
<tr>
<td>28</td>
<td>LL</td>
<td>Local to local</td>
<td>29</td>
<td>LEM</td>
<td>Loan to employee ratio</td>
</tr>
<tr>
<td>30</td>
<td>LEM</td>
<td>Loan to employment</td>
<td>31</td>
<td>LEM</td>
<td>Loan to employee ratio</td>
</tr>
<tr>
<td>32</td>
<td>MD</td>
<td>Mortgage to house</td>
<td>33</td>
<td>HOS</td>
<td>Health to hospital</td>
</tr>
</tbody>
</table>

**Attributes**

- Crime: CR75/100, CE/P
- Income: I/BH, SE/H
- Age: AL, CI
- Education: CR75/100, CE/P
- Occupation: A/BH, AL
- Census: CI, CE/P
- School: G/S, S/STU, H/SSTU
- Health: H/S, M/S, LEM
- Employment: MDR, MD, LEM, LEM
- Mortgage: M/S, LEM, MD, MD
- License: LIM, LEM, LEM, LIM
- Loan: LIM, LEM, LEM, LEM
- Health to hospital: HOS, HOS
- Mortgage to house: MDR, MD, MD
- Loan to house: LIM, LIM, LIM
- Mortgage to school: M/S, M/S
- Loan to school: LEM, LEM, LEM
- Student to urban: S/STU
- Local to local: LL
- License to operate: LEM
- Income to house ratio: I/BH
- Health to school: HOS
- Loan to employee ratio: LEM, LEM
- Loan to employee ratio: LEM
- License to employee ratio: LEM
- Mortgage to employee ratio: M/S
- Loan to employee ratio: LEM
- Mortgage to employee ratio: M/S
- Loan to employee ratio: LEM
- Health to hospital: HOS
- Health to hospital: HOS
- Income to house ratio: I/BH
Graph 3. Multilayer Perceptron on selected attributes

Graph 4. Pace Regression selected attributes

Graph 5. Weights of sigmoid node on selected attribute
Graph 6. Weights of sigmoid node on complete attribute set

Graph 7. Comparison of Multilayer Perceptron and Pace Regression on all attributes

Graph 8. Comparison of Multilayer Perceptron and Pace Regression on selected attributes
Graph 9. Comparison of Multilayer Perceptron selected and Pace Regression on all attributes

Graph 10. Comparison of Multilayer Perceptron on all and Pace Regression selected attributes

Graph 11. Comparison of Pace Regression on all and selected attributes

Graph 12. Comparison of Multilayer Perceptron on selected and on all attributes