International Journal of Research in Advent Technology, Special Issue, NCRCEST 2019
E-ISSN: 2321-9637
Available online at www.ijrat.org

User Item Recommendation System Using Machine Learning

[1]R.Viswanathan,[2]T.Edison, [3]D.N. Kumar
[1,2] School of computer science in Vels University Chennai,
[3]Professor at School of computer science in Vels University Chennai,
rvnathan06@gmail.com,edison.gis@gmail.com,kumar.se@velsuniv.ac.in,

Abstract: The need for recommendation system has increased with the rapid improvement in the use of digital solutions for every problem. The amount of digital data produced is overwhelming and difficult to manage, especially in the banking sector due to the fact that data handed is more protected and critical. The objective is to develop a recommendation system to assist the management to decide upon whether or not a customer will create a account in the bank using machine earning algorithms. Here we have tried to apply data analytics combined with machine learning algorithms to reduce the number of inputs by maintaining the accuracy of the prediction.
Key Words: Recommendation System, Machine Learning, Banking.

1. INTRODUCTION

The use of recommendation system has increased a lot in the last decade due to the improvements in the neural network and machine learning algorithm. Although the machine learning algorithms produce a decent accuracy of prediction the problem with the data available dose not end here. The real issue is when we try to decrease the number to inputs given to the system to train it. Whenever we start to train a system to make predictions the quality of data and the quantity plays an important role as it determines the accuracy of the prediction. Here we took a banking dataset which has 20 attributes to determine whether or not a individual who have been marketed will seek for the services of the bank. Here we try to predict the outcome by reducing the number of attributes and same time with a improved accuracy.

2. MACHINE LEARNING

Data scientists use many various varieties of machine learning algorithms to find patterns in huge knowledge that result in unjust insights. At a high level, these totally different algorithms is classified into two teams supported the approach they "learn" concerning knowledge to create predictions: supervised and unattended learning.

Supervised Machine Learning: the bulk of sensible machine learning uses supervised learning. supervised learning is wherever you've got input variables (x) and an output variable (Y) and you employ an algorithmic program to be told the mapping perform from the input to the output Y = f(X) . The goal is to approximate the mapping perform therefore well that once you have new information|input file|computer file} (x) that you simply will predict the output variables (Y) for that data.

Techniques of supervised Machine Learning algorithms embody linear and supply regression, multi-class classification, call Trees and support vector machines. supervised learning needs that the info wont to train the algorithmic program is already labelled with correct answers. for instance, a classification algorithmic program can learn to spot animals once being trained on a dataset of pictures that area unit properly labelled with the species of the animal and a few distinguishing characteristics.

Supervised learning issues are often more sorted into Regression and Classification issues. each issues have as goal the development of a summary model which will predict the worth of the dependent attribute from the attribute variables. The distinction between the 2 tasks is that the indisputable fact that the dependent attribute is numerical for regression and categorical for classification.

***Classification*:**

A classification algorithms main aim is to predict the output variables from the given data that whether the target variable belong to a specific class like "yes" or "no". A classification model makes an attempt to draw some conclusion from ascertained values. Given one or additional inputs a classification model can try and predict the worth of 1 or additional outcomes. For example, once filtering emails "spam" or "not spam", once staring at dealings knowledge, "fraudulent", or "authorized". The classification algorithm tries to find the class the attribute belongs to or tries to predict the class label value to which it is mapped by the training set. There are variety of classification models. Classification models embrace supply regression, call tree, random forest, gradient-boosted tree, multilayer perceptron, one-vs-rest, and Naive Bayes.

It is common for categoryification models to predict an eternal worth because the chance of a given example happiness to every output class. the chances will be taken because the probability or confidence of a given example happiness to every category. A foretold chance will be reborn into a category worth by choosing the category label that has the best chance. There are many ways to estimate the talent of a classification prognostic model, however maybe the foremost common is to calculate the classification accuracy. The classification accuracy is that the share of properly classified examples out of all predictions created.

Decision Tree Classification:

The understanding level of decision Trees rule is really easy compared with different classification algorithms. the choice tree rule tries to unravel the matter, by

mistreatment tree illustration. every internal node of the tree corresponds to associate degree attribute, and every leaf node corresponds to a category label. In call trees, for predicting a category label for a record we tend to begin from the foundation of the tree. we tend to compare the values of the foundation attribute with record's attribute. On the idea of comparison, we tend to follow the branch comparable to that worth and jump to successive node. We continue scrutiny our record's attribute worths with different internal nodes of the tree till we tend to reach a leaf node with expected category value. As we all know however the shapely call tree is accustomed predict the target category or the worth. currently let's understanding however we are able to produce the choice tree model.

It's a add of product illustration. The add of product(SOP) is additionally called appositive traditional type. For a category, each branch from the foundation of the tree to a leaf node having constant category may be a conjunction(product) of values, completely different branches ending in this category type a disjunction(sum). the first challenge within the call tree implementation is to spot that attributes can we got to take into account because the root node and every level. Handling this is often understand the attributes choice. we've got completely different attributes choice live to spot the attribute which may be thought-about because the root note at every level.

Random Forest Classification:

Random Forest may be a supervised learning algorithm. such as you will already see from it's name, it creates a forest and makes it somehow random. The "forest" it builds, is associate ensemble of call Trees, most of the time trained with the "bagging" technique. The overall plan of the sacking technique is that a mixture of learning models will increase the result. With a number of exception a random-forest classifier has all the hyperparameters of a decision-tree classifier and additionally all the hyperparameters of a sacking classifier, to manage the ensemble itself. rather than building a bagging-classifier and spending it into a decision-tree-classifier, you'll be able to simply use the random-forest categoryifier class, that is additional convenient and optimized for call trees. Note that there's additionally a random-forest regressor for regression tasks.

The random-forest rule brings additional randomness into the model, once it's growing the trees. rather than looking for the most effective feature whereas cacophonous a node, it searches for the most effective feature among a random set of options. This method creates a large diversity, that usually ends up in a more robust model. Therefore after you square measure growing a tree in random forest, solely a random set of the options is taken into account for cacophonous a node. you'll be able to even build trees additional random, by victimization random thresholds on high of it, for every feature instead of looking for the most effective doable thresholds (like a standard call tree does).

***Execution*:**

The first step is to select the data attributes to be used for decision tree algorithm. Attributes like contact information and address are removed. The preliminary observation is that it has a average class error of 26% and the overall error was 8.5 %.

```
Error matrix for the Decision Tree model on raw.csv [validate] (counts):

      Predicted
Actual  no yes Error
   no  532  21   3.8
   yes  31  33  48.4

Error matrix for the Decision Tree model on raw.csv [validate] (proportions):

      Predicted
Actual   no yes Error
   no  86.2 3.4   3.8
   yes  5.0 5.3  48.4

Overall error: 8.5%, Averaged class error: 26.1%

Rattle timestamp: 2018-03-06 23:07:23 dhaksj1
======================================================================
```

Figure 1: primary classification

By looking how the split was made in the decision we came to a conclusion that only eight attributes contribute prediction of data which is added to the trainer that both in Gini and entropy prediction methods of decision tree. By reducing the number of inputs to the decision tree the overall error rate of the prediction was increased because there is only fewer data to play with. The error rate git increased to 9.3% with a average class error of 23%.

```
Error matrix for the Decision Tree model on raw.csv [validate] (counts):

      Predicted
Actual  no yes Error
   no  529  24   4.3
   yes  33  31  51.6

Error matrix for the Decision Tree model on raw.csv [validate] (proportions):

      Predicted
Actual   no yes Error
   no  85.7 3.9   4.3
   yes  5.3 5.0  51.6

Overall error: 9.3%, Averaged class error: 27.95%

Rattle timestamp: 2018-03-07 00:02:35 dhaksj1
```

Figure 2: Classification after attribute reduction

The split with the resulting shows that only 2 attributes played an important role in the prediction those are duration of the call and number of days past after the call made and employment status. To cross compare the intuition we got we applied Pearson correlation to check the correlation between the attributes which showed there is reasonable correlation with the above attributes.

```
Correlation summary using the 'Pearson' covariance.

Note that only correlations between numeric variables are reported.

                pdays nr.employed    duration
pdays        1.00000000  0.38576457 -0.04191812
nr.employed  0.38576457  1.00000000 -0.03859648
duration    -0.04191812 -0.03859648  1.00000000

Rattle timestamp: 2018-03-07 00:04:13 dhaksj1
```

Figure 3: correlation results

To add the relatability of the data further data pruning was done to the duration attribute and the outliers were removed using boxplot algorithm. Initial 4000 tuples of data was reduced to 3000 tuples.

By adding this pre-processed and pruned data to the classifier algorithm the error of the prediction came to 8.7 % and by cross comparing the prediction with random forest algorithm the overall error was 9.6.

```
C:\Users\dhaksj1\AppData\Local\Programs\Python\Python36-32\python.exe C:/Users/dhaksj1/PycharmProjects/untitled/hai.py
Dataset Lenght:  3828
Dataset Shape:  (3828, 3)
Dataset:     duration  nr.employed   y
0         0       5191.0  no
1         4       5191.0  no
2         5       4963.6  no
3         5       4963.6  no
4         5       5228.1  no
Results Using Gini Index:
Predicted values:
['no' 'no' 'no' ... 'no' 'no' 'no']
Confusion Matrix:  [[1020   39]
 [  38   52]]
Accuracy :  93.29852045256744
Report :              precision    recall  f1-score   support

         no       0.96      0.96      0.96      1059
        yes       0.57      0.58      0.57        90

avg / total       0.93      0.93      0.93      1149
```

Figure 4: final results

3. CONCLUSION:

Thus by data pruning using decision tree split structure, Pearson correlation and boxplot the data size which used to train the classifier was reduced drastically to three attributes. The system could produce results to accuracy level of 93% and 91 % respectively through decision tree and random forest respectively.

REFERENCES

[1] Al Nuaimi, Z.N.A.M., & Abdullah, R. (2017). Neural network training using hybrid particlemove artificial bee colony algorithm for pattern classification. *Journal of Information and Communication Technology*, 16 (2), pp. 314-334.

[2] C. Thirumalai & R. Manzoor. (2017) Investigating the breast cancer tissue utilizing semi-supervised learning and similarity measure. *International conference of*

Electronics, Communication and Aerospace Technology (ICECA), pp. 269-274.

[3] C. Thirumalai & R. Manzoor. (2017). Cost optimization using normal linear regression method for breast cancer Type I skin. *International conference of Electronics, Communication and Aerospace Technology (ICECA),*pp. 264-268.

[4] C. Thirumalai & M. Senthilkumar. (2017). An assessment framework of intuitionistic fuzzy network for C2B decision making. *4th International Conference on Electronics, and Communication Systems (ICECS),* pp. 164-167.

[5] C. Thirumalai, G. V. SaiSharan, K. V. Krishna & K. J. Senapathi. (2017). Prediction of diabetes disease using control chart and cost optimization-based decision. *International Conference on Trends in Electronics and Informatics (ICEI),* pp. 996-999.

[6] Kasim, M.M., Kashim, R., & Khan, S.A.M.N. (2017). A linear programming based model to measure efficiency and effectiveness of undergraduate programs. *Journal of Information and Communication Technology,* 16 (2), pp. 394-407.

[7] K. R. Mujahid & C. Thirumalai. (2017). Pearson Correlation Coefficient Analysis (PCCA) on Adenoma carcinoma cancer. *International Conference on Trends in Electronics and Informatics (ICEI),* pp. 492-495.

[8] K. Sharma, B. Muktha, A. Rani & C. Thirumalai, (2017). Prediction of benign and malignant tumor. *International Conference on Trends in Electronics and Informatics (ICEI),* pp. 1057-1060.

[9] Mokhtar, S.A., Wan Ishak, W.H., & Norwawi, N.M. (2016). Modeling reservoir water release decision using Adaptive Neuro Fuzzy Inference System. *Journal of Information and Communication Technology,* 15 (2), pp. 141-152.

[10] Olvi L. Mangasarian, Computer Sciences Dept. University of Wisconsin, 1210 West Dayton St., Madison, WI 53706

[11] Othman, M., & Azahari, S.N.F. (2016). Deseasonalised forecasting model of rainfall distribution using fuzzy time series. *Journal of Information and Communication Technology,* 15 (2), pp. 153-169.

[12] R. Poovarasan, S. Keerthi, K. Yuvashree & C. Thirumalai. (2017). Analysis on diabetes patients using Pearson, cost optimization, control chart. *International Conference on Trends in Electronics and Informatics (ICEI),* pp. 1139-1142.

[13] Ramli, R., Jamaluddin, F., Bakar, E.M.N.E.A., Alias, M.Y., Mahat, N.I., & Karim, M.Z.A. (2013) Assignment of spectrum demands by merits via analytic hierarchy process and integer programming. *Journal of Information and Communication Technology,* 12 (1), pp. 39-53.

[14] Srisawat, C., & Payakpate, J. (2016). Comparison of MCDM methods for intercrop selection in rubber plantations. *Journal of Information and Communication Technology,* 15 (1), pp. 165-182.

[15] Vaishnavi B., K. Yarrakula, Karthikeyan J. & C. Thirumalai. (2017) An assessment framework for Precipitation decision making using AHP. *11th International Conference on Intelligent Systems and Control (ISCO),* pp. 418-421.