STUDY ON PERFORMANCE OF CLUSTERING ALGORITHMS FOR THE PREDICTION OF FERTILIZER

Dr. K. Shyamala and Mrs. I. Rajeshwari

1Associate Professor of Computer Science, Dr. Ambedkar Govt. Arts College, Chennai, Tamilnadu, India
2Research Scholar, Department of Computer Science, Dr. Ambedkar Govt. Arts College, Chennai, Tamilnadu, India

ABSTRACT

Agriculture plays a crucial role in Indian economy. It is the backbone for developing countries like India as more than 70% of population depends on agriculture. To increase crop production many factors are responsible like soil, weather, rain, fertilizers and pesticides, out of which the soil is the main key factor. To maintain nutrient levels in the soil, fertilizers are added to the soil. Choosing the needed quantity of fertilizer is one of the issues Indian farmers are facing. Providing excess or insufficient fertilizer can affect the plant life and reduce the yield. In this work, plant, soil nutrients and fertilizers data are used to increase crop production. This paper presents an approach which uses Data Mining clustering techniques to find out the fertilizers needed based on the availability of soil nutrients for the corresponding crop. Clustering algorithms like Mini batch Kmeans, Affinity propagation, Mean shift, Spectral, Ward, Agglomerative, DBSCAN and Birch are deployed on fertilizer dataset which consists of soil, plant and fertilizer particulars. The performance of these algorithms are analysed based on metrics like execution time, Silhouette coefficient and Davies-Bouldin score.

Keywords -Fertilizer, Data Mining, Clustering, Agglomerative clustering, DBSCAN.

[1] INTRODUCTION

India is an Agricultural country and the major occupation of Indians is farming. Agriculture plays a vital role in the growth of Indian economy. But the decline in the productivity is being seen for a few decades. The reasons for this declination are fragmented land holdings, lack of irrigation, illiteracy of Indian farmers, lack of knowledge about their soils and crops to be grown, decision making capacity in choosing good seeds, fertilizer etc. [1].

One of the prime sources of nutrients for crops is soil which is a natural resource and supports the growth of plant in many ways. Hence, obviously the health of soil should be maintained properly for sustaining productivity over long run. Knowledge about soil health and its maintenance is critical for sustaining crop productivity [2]. Soil testing is essential for maintaining soil health and to improve nutrient usage efficiency. Soil testing is a pre-cultivation activity which gives a brief idea about soil structure and mineral compositions rations. Surveying this reports, agriculture scientists provide guidelines for crop harvesting and fertiliser usage in harvesting. Soil testing must be done frequently to analyze the vitality of soils during the seasonal changes. Indian Government has established soil test centres at every district head quarters [3]. During testing process, the essential nutrients required for the growth of various crops can be estimated. The soil test provides information about the
various chemical properties such as EC, pH, along with micro nutrients N, P, K and macro nutrients Zn, B, Cu, Fe [4].

In this paper, data mining clustering techniques are applied over the fertilizer dataset consisting of data about the crop, soil nutrient and fertilizer, to predict the fertilizers to be applied on the soil to improve the production of the corresponding crop. The main aim of the paper is to assist farmers in improving the crop yield by applying appropriate quantity of correct fertilizer based on the soil nutrient and crop cultivated.

Clustering algorithms like Mini batch Kmeans, Affinity propagation, Mean shift, Spectral clustering, Ward, Agglomerative clustering, DBSCAN and Birch are deployed on the dataset. The performance of these algorithms are analysed based on metrics like execution time, Silhouette score and Davies-Bouldin score. The main problem is comparative study of these clustering algorithms using the fertilizer dataset.

The organization of this paper is as follows. In Section 2, the literature review conducted is described and the study to be done is introduced. Section 3 describes the dataset and the clustering techniques used. Section 4 discusses the results of the clustering algorithms for predicting the fertilizers to be applied. Finally, section 5 includes conclusion and future work.

[2] LITERATURE REVIEW

Pallavi et al. [5] analyzed and compared the behaviour of three different clustering algorithms, viz., K- Means, Farthest First and Hierarchical clustering algorithm on Wine, Haberman and Iris dataset using WEKA tool. The algorithms were compared based on computation time and performance percentage computed by taking the proportion of the correctly cluster building instances with the incorrectly formed cluster. This comparision is done before and after performing Principal Component Analysis (PCA). It was concluded that the K-means algorithm performs well before and after performing PCA, followed by Hierarchical clustering, then Farthest fast clustering. When execution time is taken into consideration, Farthest first clustering gives a fast analysis, but makes comparatively high error rate.

Oyelade et al. [6] applied Kmeans clustering algorithm on a data set of a Nigerian private school results for a semester with nine courses of 79 students for performance evaluation. The performance of the algorithm was compared with the already existing systems like fuzzy, rough set theory as a classification approach, etc., and concluded that the Kmeans clustering algorithm acts as a good benchmark to monitor the students’ performance in higher institution thereby paying a way to improve the student’s subsequent academic results.

Jignash et al. [7] reviewed the various data mining techniques used for fertilizer recommendation based on soil nutrients. In Kmeans clustering, based on Euclidean distance, object is placed into the respective cluster [8]. K-means approach can be used for classifying plant and soil, monitoring change in water quality, grading apples before marketing, weeds detection and wine fermentation prediction problems [9].

Jharna et al. [10], deployed PAM, CLARA, DBSCAN and multiple linear regression on crop, soil and climatic data to find the optimal parameters to maximize the crop production and performed the performance analysis of the algorithms. A modified DBSCAN algorithm was designed by assigning a value for K obtained from applying Batchelor Wilkins clustering algorithm. This modified DBSCAN algorithm is applied to cluster the data based on districts based on temperature, rain fall and soil type. PAM and CLARA are used to
cluster the data. Crop production can be improved by obtaining the optimal parameters based on these analyses. For predicting the annual crop yield, multiple linear regression method is used. The performance of the algorithms were analysed based on the external quality metrics like Purity, Homogeneity, Completeness, Rand Index, Precision, Recall and F measure. It was concluded that when compared to PAM and CLARA, DBSCAN gives the better clustering quality, CLARA gives the better clustering quality than the PAM.

Camila et al. [11], performed a data analysis of social information in order to identify profiles of small farmer families from the state of Ceará, using statistical and machine learning techniques from data mining and cluster analysis. The data were obtained from small producers and family farmers from six cities in Ceará state, Northeastern Brazil, through research conducted directly with the families, by means of personal questions regarding demography, economics, agricultural production, and food security. In order to work with this challenging data, a methodology capable of handling variables of mixed types and null values was deployed. The best number of clusters within the data was obtained with the Silhouette technique, and the PAM clustering algorithm was employed in order to partition the data in the clusters and found that the analyzed families could be divided into a best number of 2 clusters.

Valarmathy et al. [12], compared the performance of eight different clustering algorithms like CLARA, K-Means clustering, COWEB, Farthest first, Filtered cluster, DBSCAN, CLOPE and Expectation maximization(EM) on the student’s higher secondary data taken from UCI machine learning repository using JDK and python based on several metrics and concluded DBSCAN performs well among other algorithms on the taken dataset.

Kyle et al. [13], compared the performance of partition-based (K-Means), density-based (DBSCAN) and hierarchical (BIRCH) clustering methods on Moodle Learning Management System log data subset and concluded that the Partition-based K-Means algorithm performed better than the Hierarchical BIRCH algorithm, and the Density-based DBSCAN algorithm produced clusters with the highest Silhouette Coefficient values and the better distribution amongst the clusters for the 2-dimension and 7-dimension datasets, and the BIRCH algorithm also performed fairly well and can act as a good starting point to find cluster groups in new datasets.

From the review, it is found that most of the authors have used few techniques and analysed. In this study, Mini Batch KMeans, Affinity Propagation, Mean Shift, Spectral, Ward, Agglomerative, DBSCAN and Birch Clustering techniques are deployed on fertilizer data set and their performance are analysed based on certain metrics like Silhouette Coefficient, Davies-Bouldin index and execution time using PYHTON.

[3] RESEARCH METHODOLOGY

Data collection:

The fertilizer data set was collected from the soil test centers in Dindigul district. The features include Crop, Soil test value of N, P & K, and Recommendation of N, Urea, P2O5, Super, K2O & MOP. The collected data was preprocessed by removing unwanted data, noisy data and blank data. The data was initially coded in EXCEL and finally converted to comma delimited .csv. The various clustering algorithms were applied on the resultant reduced dataset.

Mini Batch KMeans Clustering:
Mini Batch K-means algorithm's main idea is to use small random batches of samples of a fixed size so that they can be stored in memory. In each iteration, a new random sample from the dataset is obtained and used to update the clusters and this is repeated until convergence. Each mini batch updates the clusters using a convex combination of the values of the prototypes and the samples, applying a learning rate that decreases with the number of iterations. This learning rate is the inverse of number of samples assigned to a cluster during the process. As the number of iterations increases, the effect of new examples is reduced, so convergence can be detected when no changes in the clusters occur in several consecutive iterations [14].

**Affinity Propagation Clustering:**

Affinity Propagation creates clusters by sending messages between data points until convergence [15]. It uses the two important parameters, preference and damping factor. The preference parameter, controls the number of exemplars (or prototypes) used, and the damping factor, damps the responsibility and availability of messages. A dataset is described using a small number of exemplars. The exemplar sample is represented by the messages sent between pairs and is updated in response to the values from other pairs. This updating happens iteratively until convergence, at that point the final exemplars are chosen.

**Mean Shift Clustering:**

Mean shift associates each data point with the nearby peak of the dataset’s probability density function. It defines a window around each data point and computes the mean of the data point. Then it shifts the center of the window to the mean and repeats the algorithm till it converges [16].

**Spectral Clustering:**

Spectral clustering uses information from the eigen values of special matrices built from the graph or the data set. Spectral clustering is a technique with roots in graph theory, where the approach is used to identify communities of nodes in a graph based on the edges connecting them. [17].

**Ward’s Hierarchical Clustering:**

Ward’s method starts with n object, each in a cluster, which are combined gradually thereby resulting in one cluster containing all objects. At each step, the process makes a new cluster that minimizes variance, measured by an index called E (also called the sum of squares index) [18].

**Agglomerative Clustering:**

Agglomerative clustering works in a “bottom-up” manner. Here, initially the number of clusters is equal to the number of object, each cluster having an object which is taken as leaf. Then gradually, the most similar two clusters are combined into a new bigger cluster (nodes). This procedure is iterated until all points are member of just one single big cluster (root) [19].

**DBSCAN clustering:**

Density-Based Spatial Clustering of Applications with Noise (DBSCAN) uses the concept of density reachability and density connectivity [20] and works well even with noisy datasets. It results in dense arbitrarily shaped clusters by expanding after finding the dense
areas. Two main parameters to DBSCAN are radius of the ‘neighborhood region’ (\(\varepsilon\)) and the minimum number of points (minPoints).

**Birch Clustering:**

BIRCH is local in that each clustering decision is made without scanning all data points or all currently existing clusters. It uses measurements that reflect the natural closeness of points, and at the same time, can be incrementally maintained during the clustering process [21].

**Silhouette Coefficient:**

The silhouette value is a measure that shows how an object is similar to its own cluster (cohesion) when compared to other clusters (separation). The separation distance between the resulting clusters can be found out [22]. Its value is between -1 and 1. 1 indicates that the sample is well-clustered, 0 indicates overlapping clusters and -1 indicates that the sample is misclassified.

**Davies-Bouldin Index:**

The Davies-Bouldin Index evaluates intra-cluster similarity and inter-cluster differences [23]. Zero is the lowest possible score. Values closer to zero indicate a better partition [24].

**[4] RESULT AND DISCUSSIONS**

The algorithms were executed on the processed dataset in .CSV (Comma Separated format) file using PYTHON and the various metrics were analysed. Table 4.1. shows the Silhouette coefficient, Davies-Bouldin Index and execution time resulted by the various algorithms applied on the processed dataset. Figures 4.1, 4.2 and 4.3 shows the comparison of the algorithms based on these metrics respectively. From table 4.1 and figure 4.1, it is inferred that DBSCAN is the fastest algorithm, followed by Birch, Mini Batch KMeans and Agglomerative clustering algorithm. A good algorithm should have Silhouette Coefficient around 1 and Davies-Bouldin Index should be around zero. From table 4.1 and figure 4.2, it is clear that Agglomerative clustering has the best Silhouette Coefficient among the 8 algorithms, followed by Mean Shift, Mini Batch KMeans and Ward. Table 4.1 and figure 4.3, indicate that Agglomerative clustering has the best Davies-Bouldin Index, followed by Mean Shift, DBSCAN and Mini Batch KMeans. Table 4.2 shows the performance of the algorithms sort out based on the metrics.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Execution Time in ms</th>
<th>Silhouette Coefficient</th>
<th>Davies-Bouldin Index</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mini Batch KMeans</td>
<td>0.028925896</td>
<td>0.4329074</td>
<td>0.8530945</td>
</tr>
<tr>
<td>Affinity Propagation</td>
<td>0.203343153</td>
<td>0.3300605</td>
<td>0.906781</td>
</tr>
<tr>
<td>Mean Shift</td>
<td>0.125650167</td>
<td>0.482126</td>
<td>0.7037594</td>
</tr>
</tbody>
</table>

**Table 4.1. Performance of the algorithms based on various accuracies**
Spectral | 0.071977854 | 0.2884665 | 1.3343721
Ward | 0.046862125 | 0.4124636 | 0.8730299
Agglomerative | 0.040266514 | 0.5059883 | 0.6413022
DBSCAN | 0.006016016 | -0.0620094 | 0.7869418
Birch | 0.016916275 | 0.3795068 | 0.9567836

**Figure 4.1.** Comparison of algorithms based on the execution time.

**Figure 4.2.** Comparison of algorithms based on the Silhouette coefficient.
Figure 4.3. Comparison of algorithms based on the Davies-Bouldin Index

Table 4.2. Algorithms sorted based on the performance from best to worst

<table>
<thead>
<tr>
<th>Execution Time</th>
<th>Silhouette Coefficient</th>
<th>Davies-Bouldin Index</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBSCAN</td>
<td>Agglomerative</td>
<td>Agglomerative</td>
</tr>
<tr>
<td>Birch</td>
<td>Mean Shift</td>
<td>Mean Shift</td>
</tr>
<tr>
<td>Mini Batch KMeans</td>
<td>Mini Batch KMeans</td>
<td>DBSCAN</td>
</tr>
<tr>
<td>Agglomerative</td>
<td>Ward</td>
<td>Mini Batch KMeans</td>
</tr>
<tr>
<td>Ward</td>
<td>Birch</td>
<td>Ward</td>
</tr>
<tr>
<td>Spectral</td>
<td>Affinity Propagation</td>
<td>Affinity Propagation</td>
</tr>
<tr>
<td>Mean Shift</td>
<td>Spectral</td>
<td>Birch</td>
</tr>
<tr>
<td>Affinity Propagation</td>
<td>DBSCAN</td>
<td>Spectral</td>
</tr>
</tbody>
</table>

[5] CONCLUSION AND FUTURE WORK

This study has conducted a comparison between the various clustering algorithms like Mini batch Kmeans, Affinity propagation, Mean shift, Spectral clustering, Ward, Agglomerative clustering, DBSCAN and Birch on fertilizer dataset using PYTHON. The algorithms are compared based on the metrics Silhouette Coefficient, Davies-Bouldin Index and Execution Time. It is concluded that Agglomerative clustering is found suitable for this dataset with little compromise in the execution time for finding the fertilizers to be applied. The other algorithms that top in the rank are Mean Shift and Mini Batch KMeans. When execution time alone is given importance, DBSCAN can be used. It is concluded that Agglomerative clustering is the better one for the fertilizer dataset and the future work is to analyze this algorithm.
REFERENCES


[23] https://www.researchgate.net/post/How_can_we_say_that_a_clustering_quality_measure_is_good