IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

HOMOGENEOUS CLUSTERING FOR DATASET REDUCTION

[1]Atharva A. More, [2]Shreyash S. Lanjewar, [3] Prof. Dr. S. M. Kamlapur, [4]Siddhesh M. Patankar, [5]Ketan P. Patil

[1,2,4,5] Students, Department of Computer Engineering

[3]Professor, Department of Computer Engineering

K. K. Wagh Institute of Engineering Education and Research, Nashik, India Affiliated under Savitribai Phule Pune University

Abstract: Big data has caused an exponential growth in data generation, making it possible that traditional data processing tools and methodologies won't be able to handle the sheer amount of data. Working with large datasets presents a number of issues, including high processing costs and storage needs. Furthermore, it can take a lot of time and computing power to analyze and extract insights from such huge databases, making it challenging for academics, analysts, and organizations to get usable information from data. A method for easing the difficulties of working with massive data is data reduction. It entails minimizing the amount of data while keeping the pertinent data required for analysis. Data reduction strategies are created to retain the accuracy and quality of the data while maximizing the usage of processing power and storage space. Reduction by homogeneous clusters is a useful technique for data reduction. Groups of data points that fall into the same class or category are referred to as homogeneous clusters. The goal of homogeneous clustering is to put together data items with comparable features, such as size, shape, colour, or other characteristics, in one group. The dataset may be compressed without losing the crucial details required for analysis by grouping data points into homogenous groups. The cheap preprocessing cost of homogeneous clustering is one of its main benefits. This indicates that preprocessing the data before clustering involves just a little amount of time and resources. A variety of datasets, including those with high-dimensional data, can use homogeneous clustering. To find the best method suitable for dataset reduction, it is necessary to compare all available dataset reduction methods and identify the most appropriate one.

***Index Terms* - Data reduction, Homogeneous clustering, Computational cost, Storage requirement, K-means clustering.**

I. Introduction

Dataset reduction is the process of minimizing the size of a dataset by deleting unnecessary, redundant, or noisy data points while keeping the crucial information. This enhances the efficacy and efficiency of data processing activities, particularly for large data applications. Dataset reduction using homogeneous clustering is an essential technique for managing and analyzing large and complex datasets efficiently. Homogeneous clustering refers to the grouping of data points that share similar features or properties, which helps in reducing the size of the dataset without losing crucial information necessary for analysis. This technique is particularly useful for high-dimensional datasets, where traditional data reduction methods might not be effective. Homogeneous clustering can be used to group data points based on a range of attributes, such as color, size, shape, or other properties, enabling more efficient and accurate analysis of large datasets.

One of the primary benefits of homogeneous clustering is that it is a low-cost and efficient preprocessing method. This means that it requires minimal time and resources to cluster the data, making it a preferred choice for many organizations dealing with large and complex datasets. Additionally, by reducing the size of the dataset, homogeneous clustering can lead to significant savings in storage and processing costs. Homogeneous clustering finds its applications in various domains, including image and video processing, bioinformatics, web search, and social network analysis.

One method of dataset reduction using homogenous clustering is the Geometrical Homogeneous Clustering for Image Dataset Reduction (GHCIDR). In this method the cluster is divided into annular areas (concentric regions of cluster) which then chooses images from each region. It gives a sense of the complete cluster by choosing images that are evenly separated from the centroid. The idea behind GHCIDR is that by sampling over the cluster's volume, representative and varied samples may be obtained. The closest point to the centroid is chosen, creating the impression that it is center and represents the whole cluster. GHCIDR evenly chooses datapoints from the cluster's whole volume. It includes all of the cluster characteristics in this way. In this paper a variation of GHCIDR is used for dataset reduction which differs form the original method by generation of annulus and selection of data points from this annulus.

Another three variations are proposed in which all the proposed methods differ in the selection of data points after applying K Means algorithm on the dataset. For dimensionality reduction, Principal Component Analysis (PCA) is also used to analyze the outcomes of the aforementioned methods.

The objectives section of the paper aims to propose a method for dataset reduction in machine learning that can successfully shrink the data set size without compromising the model's performance quality. It explains the importance of dataset reduction and its potential impact on model accuracy.

The literature section of the paper covers various topics related to data reduction, clustering algorithms, and instance-based learning. It includes discussions on Geometrical Homogenous Clustering for Image Dataset Reduction, KEEL, lightweight coresets, k-means algorithm, uniform deviation bounds, prototype selection, and a new clustering method based on k-means. The papers provide different techniques for reducing data sets while preserving relevant information and improving clustering accuracy. They also offer solutions for initializing algorithms, addressing noise and outliers, and improving the trade-off between accuracy and reduction.

The methodology section provides the four different approaches for reducing the size and improving the homogeneity of a dataset. The first approach involved using K-Means clustering to group similar data points together and selecting representative subsets from each cluster using an annulus-based method. The second approach used a threshold based on the maximum class count for each cluster to select data points with higher individual class count. The third approach involved setting a threshold based on the maximum distance of a data point from the center of the cluster, resulting in a reduced dataset with only the most representative data points. The fourth approach used a recursive method for homogeneous clustering to produce homogeneous clusters without removing any data points from the dataset, followed by a datapoint selection technique to refine the dataset. These approaches can be used to reduce the size of large datasets and make them more homogeneous for easier analysis.

The results section presents the findings of the study, using tables, figures, and charts to illustrate the data analysis. It provides a clear and concise summary of the experimentation. The conclusion summarizes the main findings of the study and restates the research question.

II. Objectives

The primary objective of this experiment is to evaluate the effectiveness of various dataset reduction techniques while ensuring that the machine learning model's accuracy is not threatened. To do this, the available methods should be examined for data set reduction and a method should be suggested that can successfully shrink the data set size without compromising the model's performance quality.

Data set reduction is crucial in machine learning because it can shorten the time and computational resources needed to train the model. The accuracy of the machine learning model may be impacted, if crucial data is lost when the size of the data set is reduced.

To solve this problem, there is a need to concentrate on creating a novel method that can shrink the size of the data collection without sacrificing any significant information. This method will be created by researching current methods and determining where advancements can be made.

III. Literature Survey

This section covers a broad review of dataset reduction techniques, use of clustering algorithms for dataset reduction in particular, the formation of homogenous clusters.

The Geometrical Homogenous Clustering for Image Dataset Reduction model selected tuples based on the intuition that can give representative and diverse samples by sampling throughout the volume of the cluster. GHCIDR performed better than the proposed approaches, random baseline and the original RHC. It may be prone to outliers (more boundary points in the reduced set). -Reduction is less in comparison to RHC because of more images selected from each cluster. [1]

With access to a database of data sets and support for the application of evolutionary learning and soft computing techniques, KEEL appears to be an invaluable tool for researchers working on data mining difficulties. Researchers who want to compare the outcomes of their methods with those of other researchers in the field will probably find the addition of the KEEL-dataset and the instructions for integrating new algorithms in KEEL to be useful. Additionally, the statistical processes module is probably going to give researchers a helpful tool to compare the outcomes of their investigations and encourage to learn that the publication includes a case study that fully applies the experimental analysis methodology and uses KEEL to show its use.[2]

The idea of lightweight coresets, which allows for both multiplicative and additive mistakes, sounds like an intriguing breakthrough. Coresets are a useful tool for scaling up clustering models to enormous data sets. It's wonderful to see that the paper offers a single approach to create lightweight coresets for soft and hard Bregman clustering as well as k-means clustering. Researchers that want to scale up their clustering models are likely to find the technique attractive because it is quicker and can create smaller coresets than current builds. The fact that the suggested method readily generalizes to statistical k-means clustering and can be used to calculate smaller summaries for empirical risk minimization is also intriguing.[3]

In fact, the k-means algorithm is a well-liked clustering method utilized in a variety of applications, including customer segmentation, data mining, and image segmentation. A set of data points is divided into k clusters, and each point is assigned to the cluster with the closest mean in order for the system to function. Minimizing the sum of squared distances between points and their cluster centers is the goal. The performance of the k-means algorithm can, however, be considerably impacted by how it is initialized. In particular, the quality of the clustering and the algorithm's speed of convergence can be influenced by the initial seed points. The phrase "the algorithm is (log k)-competitive with the optimal clustering" denotes that the algorithm performs no worse than the optimal clustering, at most by a constant factor.[4]

In order to find uniform deviation bounds for loss functions that are unbounded, the paper introduces a unique approach. Because many popular loss functions, such the squared loss, are unbounded, this is significant. The framework enables the development of competitive uniform deviation bounds for the well-known clustering technique k-Means. The authors demonstrate that the rate of the uniform deviation bound can be increased from O(m(-1/4)) to O(m(-1/2)) given lax assumptions on the underlying distribution, such as a constrained fourth moment. As a result, the bound is tighter and gives a better indication of how well the model is doing. Additionally, they demonstrate that the distribution's kurtosis, which gauges the "tailenders" of the data, affects the rate at which the uniform deviation bound is reached.[5]

A new fast non-parametric algorithm for data reduction which works by using the k-Means algorithm to recursively cluster the training dataset into homogeneous clusters. The condensed set consists of the centroids of the final clusters. RHC also ignores the size of clusters and selects a single image (centroid), making the contribution of small and large clusters to be equal in the reduced dataset.[6]

Instance-based learning, a unique framework for supervised learning that makes classification predictions using individual examples, is introduced in the study. Instance-based learning does not maintain such abstractions, in contrast to standard algorithms that learn from abstractions created from instances, potentially resulting in less storage requirements. The authors explain how instance-based learning may be implemented using an extension to the closest neighbor method that greatly reduces storage needs without compromising learning rate or classification accuracy. They also demonstrate the effectiveness of this method on various real-world databases. However, the authors point out that when attribute noise is present in training examples, the storage-reducing algorithm's effectiveness declines quickly. The paper suggested modifying the algorithm to add a significance test to separate noisy cases in order to improve performance.[7]

Prototype Selection by Clustering (PSC) method for prototype selection method selects border prototypes (in order to preserve the class discrimination regions) and some interior prototypes. Due to the recursive nature, the size of the reduced dataset is not determined; hence, this makes it difficult for the user to get a better trade-off between accuracy and reduction [8]

Representation of a new clustering method based on k-means that have avoided alternative randomness of initial center. This paper focused on K-means algorithm to the initial value of the dependence of k selected from the aspects of the algorithm is improved.[9]

IV. METHODOLOGY

The first approach involves applying the K-Means clustering algorithm to the dataset to group similar data points together. Once the clusters are formed, the datapoints with the highest class count for each cluster are selected, and a representative subset of datapoints is chosen from each cluster using an annulus-based method. This results in a condensed dataset that is more homogeneous and easier to analyze. The second approach is similar to the first, but instead of selecting datapoints based on their class count, a threshold is set based on the maximum class count for each cluster. Only those datapoints whose individual class count is higher than this threshold value is selected from each cluster, resulting in a reduced dataset with a good balance of class representation. The third approach involves setting a threshold value based on the maximum distance of a datapoint from the center of the cluster. Only those datapoints within the threshold range are retained, while outliers are removed. This results in a reduced dataset with only the most representative datapoints. The fourth approach involves applying a recursive method for homogeneous clustering to produce homogeneous clusters without removing any datapoints from the dataset. After achieving homogeneous clusters, a datapoint selection technique is used to refine the dataset. This approach involves creating annuluses within each cluster and selecting datapoints uniformly from each annulus to create a representative subset of datapoints for each cluster.

Method 1: Selection of datapoints with maximum class count for every cluster

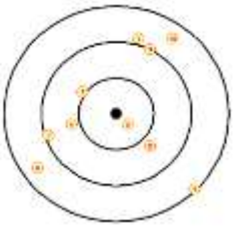
In this approach the number of datapoints in each class for each cluster are determined that the K means algorithm produces. The next step is to delete the other datapoints from each cluster and choose the datapoints with the highest class count for each cluster. This process results in homogeneous clusters that are easier to analyze and interpret.

Once the clusters are homogeneous, the datapoint selection method is applied for every cluster by forming β annuluses (whose value depends on the reduction parameter α, size and radius of the cluster) inside the clusters and selecting only those datapoints which are farthest apart from the centroid for every annulus till 4/5 of the radius of the cluster and for the 1/5 part of the cluster selecting the datapoints in the same way by keeping the value of β equal to 1, which further refine the dataset and improve the performance. This involves selecting a representative subset of datapoints from each cluster to preserve the essential information while reducing the overall size of the dataset.

Algorithm:

- Apply K Means Clustering algorithm to the dataset by assuming the number of clusters formed equal to the number of classes.
- For each cluster find the number of datapoints of each class and find the class with maximum count for every cluster.
- Select the datapoints with the highest class count from each cluster which makes the cluster homogeneous.
- Apply the datapoint selection algorithm on the homogeneous clusters formed, thus the condensed dataset is obtained.

$$\beta = \frac{maxLength}{((1 - \alpha) \times (size\ of\ cluster))}$$



Annulus of a cluster

Method 2: Selection of datapoints having class count greater than threshold

In this approach, clustering is done on the dataset to group similar data points together. Once the clustering is done, only those datapoints are selected from each cluster whose individual class count is higher than a certain threshold. This threshold value is set to be equal to 25% of the maximum class count for each cluster.

This approach allows to reduce the size of the dataset by only keeping the most representative data points from each cluster, while still maintaining a good balance of class representation. By choosing a threshold based on the maximum class count, it ensures that only those datapoints are selected that are highly representative of the underlying distribution of the data within each cluster.

The resulting clusters are likely to be heterogeneous, which means that they may contain data points from multiple classes. However, this is not necessarily a problem, as the main goal of dataset reduction is to preserve the overall distribution of the data while reducing its size. In fact, having heterogeneous clusters can be beneficial, as it can provide a more diverse set of data points for downstream tasks such as classification.

The second approach described here provides a simple and effective approach for reducing the size of a dataset while maintaining a good balance of class representation. By selecting only the most representative data points from each cluster, it can help significantly to reduce the computational complexity of downstream tasks without sacrificing too much information.

Algorithm:

- Apply K Means Clustering algorithm to the dataset by assuming the number of clusters formed equal to the number of classes.
- For each cluster find the number of datapoints of each class and find the class with maximum count for every cluster.
- For every cluster assign a threshold value equal to 25% of the maximum class count.
- Select the datapoints with the class count greater than or equal to the threshold value from each cluster which makes the cluster heterogeneous.
- Thus, forming the reduced dataset.

Method 3: Selection of datapoints with respect to distance from the centroid

The third approach involves setting a threshold value based on the maximum distance of a datapoint from the center of the cluster. This threshold value depends on the maximum distance of a datapoint belonging to the class with maximum class count. 0This threshold value serves as a way to identify the range of distances from the centroid within which the datapoints should be retained.

After identifying this threshold range, the datapoints falling outside the range are excluded, leaving only those within the threshold range. This step is important as it ensures that only the most representative datapoints are included in the cluster. By removing the outliers, the overall quality and accuracy of the clustering results are improved.

Algorithm:

- Apply K Means Clustering algorithm to the dataset by assuming the number of clusters formed equal to the number of classes.
- For each cluster find the number of datapoints of each class and find the class with maximum count for every cluster.
- Find the maximum distance of the datapoint belonging to the maximum class count from the centroid for each cluster and assign the threshold value equal to this maximum distance.
- Select the datapoints having the distance from the centroid less than or equal to the corresponding threshold value for each cluster.
- Thus, forming the reduced dataset.

Method 4: Applying homogeneous clustering and datapoint selection algorithm

In the fourth approach, recursive method has been applied for homogeneous clustering over the dataset to produce homogeneous clusters without removal of any datapoint from the dataset. After the formation of the homogeneous clusters, next the datapoint selection algorithm is applied based on the value of β(number of annuluses formed), α(reduction parameter), radius of the cluster (distance of the datapoint which is farthest from the centroid) and size of the cluster (number of datapoints within the cluster). Datapoints are chosen uniformly from the entire volume of the cluster.

After achieving homogeneous clusters, the datapoint selection technique is utilized to refine the dataset. This approach involves creating β annuluses within each cluster, where β is determined by the reduction parameter α, as well as the size and radius of the cluster. Within each annulus, select only the datapoints that are farthest away from the centroid, up to a distance of 4/5 of the cluster's radius. For the remaining 1/5 portion of the cluster, select datapoints in the same way but with a fixed β value of 1. By implementing this technique, it can effectively reduce the size of the dataset while still maintaining essential information, resulting in improved performance. Essentially, choosing a representative subset of datapoints from each cluster to achieve this goal.

Algorithm:

- Apply K Means Clustering algorithm to the dataset by assigning the number of clusters formed equal to the number of classes.
- Identify the heterogeneous clusters formed and apply K Means Clustering algorithm on the datapoints of the heterogeneous clusters by assigning the number of clusters formed equal to the number of distinct classes in the heterogeneous cluster.
- Repeat step 2 until all the clusters formed are homogeneous.
- After the formation of homogeneous clusters, apply the selection algorithm on each homogeneous cluster by forming β annuluses for every cluster based on the values of α, cluster radius and size of cluster as mentioned above. From each annulus select the datapoint having the farthest distance from the centroid. Thus, forming the condensed dataset.

V. RESULTS AND DISCUSSION

The experimentation is done on two datasets MNIST and Wine dataset.

DATASETS:

MNIST: The MNIST dataset is a popular dataset in machine learning, computer vision, and pattern recognition. It consists of 70,000 handwritten digits that are stored as 28x28 pixel images. The dataset is divided into a training set of 60,000 images and a test set of 10,000 images. Each image is labelled with the corresponding digit (0-9) it represents. The MNIST dataset is commonly used for training and testing machine learning algorithms that aim to recognize and classify handwritten digits. It is a standard benchmark dataset that is used to evaluate the performance of various machine learning models and algorithms.

Wine dataset: The Wine dataset is another popular dataset that is commonly used in machine learning and pattern recognition. It consists of the results of a chemical analysis of wines grown in a specific area of Italy. The dataset contains 13 different attributes, including alcohol content, acidity, and various chemical concentrations. The dataset contains data for three different classes of wine, each representing a different cultivar. The Wine dataset is often used for classification tasks, where the goal is to predict the cultivar of the wine based on its chemical properties. It is also used for clustering tasks, where the goal is to group similar wines together based on their chemical properties.

Results of Descriptive Statics of Study Variables

	MNIST				WINE			
Methods	%Accuracy	%Reduction	%Accuracy with PCA	%Reduction with PCA	%Accuracy	%Reduction	%Accuracy with PCA	%Reduction with PCA
Full dataset	91.869%	-	91.910	-	50.297	-	51.692	-
Method 1	68.020	96.533	44.652	98.908	7.569	97.844	7.345	99.212
Method 2	89.751	15.898	87.155	15.060	51.528	30.060	51.569	31.953
Method 3	92.432	0.083	91.880	0.031	52.842	62.305	54.338	51.562
Method 4	90.074	87.545	88.749	77.156	51.508	36.847	52.430	49.191

The table shows the accuracy results of four different methods applied to the MNIST and Wine datasets, with and without dimensionality reduction using PCA.

ANALYSIS:

MNIST: The full dataset achieved an accuracy of 91.869%, which can be considered as a baseline. Method 1 achieved the lowest accuracy of 68.02%, which is significantly worse than the full dataset but reduced the size of the dataset by 96.533%. Method 2 reduced the size by 15.898%, and the accuracy dropped slightly to 89.75%. Method 3 achieved the highest accuracy of 92.43%, which is slightly better than the full dataset but only had a negligible reduction in size of dataset. Method 4 achieved an accuracy of 90.070%, which is close to the full dataset and significantly reducing the size of dataset by 87.545%. Overall, the performance of these methods on the MNIST dataset is impressive for Method 4. The full dataset achieved the highest accuracy, and the modified Methods 1 and 3 did not achieve significant improvements. Method 3 achieved a slightly better accuracy, but the improvement is marginal.

WINE: The full dataset achieved a low accuracy of 50.297%, which is close to random guessing. Method 1 reduced the dataset up to 97.844% with least accuracy, which is not very impressive. Method 2 and Method 4 also achieved higher accuracies than the full dataset, although the improvements are not as dramatic. While there is a significant reduction in dataset size in Method 2. Method 3 achieved an accuracy of 52.842%, which is higher than the full dataset, but the improvement is not as significant as Method 1. The performance of these methods on the Wine dataset is more interesting. The full dataset achieved a low accuracy, indicating that this task is challenging. Method 2 and Method 4 also achieved higher accuracies, but the improvements are not as dramatic. Method 3 also achieved a higher accuracy, but the dataset reduction was effective.

VI. CONCLUSION

The experimentation highlights that PCA is a powerful technique for reducing the dimensionality of datasets while still maintaining acceptable levels of accuracy. Dimensionality reduction is essential in data analysis because it can help to simplify complex data and reduce the risk of overfitting, thereby improving the performance of machine learning algorithms. However, the table also demonstrates that the effectiveness of PCA in dimensionality reduction can vary depending on the specific method used and the characteristics of the dataset being analyzed. Some methods may produce more significant reductions in dimensionality but sacrifice accuracy, while others may preserve accuracy but result in smaller reductions. Proposed Method 4 emerges as a consistent performer in the table, demonstrating strong results on both datasets. This finding suggests that Method 4 may be a valuable approach to consider for dimensionality reduction, as it strikes a balance between reduction in dimensionality and accuracy.

VII. REFERENCES

[1] Shril Mody ,Janvi Thakkar, and Devvrat Joshi,Siddharth Soni, “Geometrical Homogeneous Clustering for Image Data Reduction” arXiv: 2208.13079v1 [cs.LG] 27 Aug 2022.

[2] Alcala-Fdez, J., Fern ´ andez, A., Luengo, J., Derrac, J., ´Garc´ıa, S., Sanchez, L., and Herrera, F. Keel data-mining ´software tool: data set repository, integration of algo-rithms and experimental analysis framework. Journal of Multiple-Valued Logic & Soft Computing, 17, 2011.

[3] Bachem, O., Lucic, M., and Krause, A. Scalable k-means clustering via lightweight coresets. In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 1119–1127, 2018.

[4] David Arthur and Sergei Vassilvitskii. 2007. k-means++: The Advantages of Careful Seeding. In Symposium on Discrete Algorithms.

[5] Olivier Bachem, Mario Lucic, S. Hamed Hassani, and Andreas Krause. 2017. Uniform Deviation Bounds for k-Means Clustering. In International Conference.
on Machine Learning.

[6] Ougiaroglou, S. and Evangelidis, G. Efficient dataset size reduction by finding homogeneous clusters. In Proceedings of the Fifth Balkan Conference in Informatics, pp. 168–173, 2012.

[7] D. W. Aha, D. F. Kibler, and M. K. Albert. Instance-based learning algorithms. Machine Learning, 6:37–66, 1991.

[8] Olvera-Lopez, J. A., Carrasco-Ochoa, J. A., and Mart ´ ´ınezTrinidad, J. F. A new fast prototype selection method based clustering. Pattern Analysis and Applications, 13(2):131–141, 2010

[9]. Chen, Z. and Xia, S. K-means clustering algorithm with improved initial center. In 2009 Second International Workshop on Knowledge Discovery and Data Mining, pp. 790–792. IEEE, 2009.