National Research University
«Higher School of Economics»
Faculty of Computer Science
Educational programme HSE University and University of London Double Degree Programme in Data Science and Business Analytics
Undergraduate
01.03.02 Applied Mathematics and Computer Science

Report

On Academic Internship Results of

Experiments with Recent Methods for Determining the Number of Clusters in K-means Algorithm

in Department of Data Analysis and Artificial Intelligence

(indicate the name of the CS Faculty Department, where internship was conducted)

Fulfilled by the student of the group #212

Zakharov Artem Alexandrovich

(Surname, First name, Patronymic, if any)

(Signature)

Checked by:
Faculty's Internship Supervisor:

Majid Sohrabi

(Surname, First name, Patronymic, if any)

Assistant, Resaerch Assistant, Department of Data Analysis and Artificial Intelligence

(CS Faculty Department, Position, Academic title if any)

21/07/2023	9	
(Date of check)	*(Grade according to 10-point scale)*	*(Signature)*

Moscow, 2023

Table of Content

Abstract

One of the main challenges with clustering algorithms is finding the optimal number of clusters, for which the set of objects will be distributed. This work is aimed at analysis of various methods of finding the optimal number of clusters for the K-Means clustering algorithm. Novel methods along with the credible ones like Elbow and Silhouette, were implemented, in order to construct a stable comparison by different parameters.

Key Words

1) **K-means algorithm**: A simple unsupervised machine learning algorithm that groups a dataset into 'k' number of clusters. The 'k' is decided by the user.
2) **Clustering**: The task of dividing the population or data points into a number of groups such that data points in the same groups are more similar to other data points in the same group and dissimilar to the data points in other groups.
3) **DWCNK**: Determine Cluster Number Without Clustering for Every K Based on Ratio of Variance to Range Algorithm.
4) **Euclidean distance**: A measure of the true straight line distance between two points in a plane.
5) **PCA**: Principal Component Analysis. A method used to reduce the dimensionality of large data sets by transforming to a new set of variables.
6) **Dendrogram**: A tree-like diagram that records the sequences of merges or splits in hierarchical clustering.
7) **Variance**: A statistical measurement of the spread between numbers in a data set. More specifically, variance measures the dispersion of a set of data points around their mean.
8) **Synthetic data generator**: A mechanism or program used to create artificial data that mimic the properties of real-world data.

1. Introduction

The essential effort of segmenting datasets into distinct, identifiable groups, known as 'clusters', forms the backbone of exploratory data analysis, pattern recognition, and various machine learning implementations. This clustering procedure aims to assemble data points that are more congruent with one another within a cluster than with data points inhabiting different clusters, based on a set of defined criteria.

A predominant technique used for clustering is the K-means algorithm, prized for its straightforwardness and computational agility. Nevertheless, this method poses a challenge: the determination of the optimal number of clusters, 'k', a parameter with a notable influence on the resultant clustering.

Traditional techniques such as the Elbow criterion have been employed as a solution to discern the most suitable number of clusters. By plotting the percentage of variance explained against the number of clusters, an 'elbow' or inflection point in the graph signifies the optimal number of clusters. Despite its popularity, the Elbow method has its limitations and often lacks clarity, leading potentially to a suboptimal cluster determination.

Recently, a new generation of methods has emerged, promising to enhance the process of determining 'k'. Grounded in diverse principles, these cutting-edge approaches introduce a higher level of complexity, potentially yielding more precise and consistent results. Therefore, comprehending, implementing, and evaluating these novel techniques are essential to assess their efficiency and applicability across different scenarios.

The focus of this project lies in investigating and contrasting several such innovative methods against the traditional Elbow criterion, specifically within the context of the K-means clustering algorithm. Through implementation within a coding environment, these methods are put to the test using a variety of real-world datasets sourced from the Irvine Repository. We further develop a synthetic data generator, allowing the production of controlled datasets and thus enabling more reliable and accurate testing of these methods.

The findings of this research endeavor are poised to offer valuable insights into the performance of the Elbow criterion in comparison with these novel methods, assisting in identifying the most efficacious approach for determining the optimal number of clusters. Further, this project may augment the wider comprehension of cluster analysis in the field of data science, offering the potential for more robust solutions in an array of data-driven tasks.

1.1 Background and Related Works

During the practice, a variety of novel cluster validity indices have been investigated. For the novel algorithms: the “Iterative Anomalous Clusters” approach, which was developed in 2021 by HSE student, “DWCNK” method was advised in 2022, “Gaussian Likelihood Score” method which was developed in 2019, “Agglomerative” was advised in 2021, “Data Depth” method was published in 2019. And for the classical methods: Elbow and Silhouette methods were chosen.

2. Objective of Internship

The primary objective of this internship is to conduct a thorough research study on recent methods for determining the optimal number of clusters in the K-means algorithm. We will learn, implement, and compare these methods, primarily focusing on the traditional Elbow criterion and a selected novel method. Through hands-on experience with coding these methods and testing them on both real-world and synthetic datasets, we aim to acquire a comprehensive understanding of their performance and utility.

3. Purpose of Internship

The purpose of this internship is threefold:

1) **Educational:** To obtain a practical, in-depth understanding of cluster analysis and its complexities, specifically in the context of the K-means algorithm. This hands-on experience will allow us to learn advanced techniques in data science, such as the creation of synthetic data generators and the implementation of different clustering methods.
2) **Research:** To contribute to the current body of knowledge on cluster determination in K-means algorithm. By testing and comparing the Elbow method and the selected novel method on various datasets, our research will offer valuable insights into their effectiveness, thereby informing future work in this field.
3) **Career Development:** To equip with valuable skills that are highly sought after in the field of data science. Through the implementation and testing of these clustering methods, we will gain experience in coding, data analysis, and problem-solving, thereby significantly enhancing their employability in the data-driven industry.

4. Calendar Schedule

№	Calendar Period	Plan of Work	Supervisor's mark on the point fulfillment (signature)
1	04.07.2023	Studying the K-means method and Elbow criterion for selection of the number of clusters.	9
2	05.07.2023	Studying the novel methods for selection of the number of clusters.	9
3	06.07.2023	Coding novel and Elbow methods.	9
4	09.07.2023	Testing novel and Elbow methods over real-world data from Irvine Repository.	9
5	12.07.2023	Coding a synthetic data generator.	8
6	15.07.2023	Testing novel and Elbow methods on synthetic data.	8
7	18.07.2023	Writing report and code.	8

5. Data Generation & Collection

The data generation process employed for this study is twofold: synthetic data and real-world data collection. For the synthetic data, a variety of two-dimensional datasets with different numbers of clusters were generated, to simulate a broad range of data distributions and characteristics. Various parameters such as number of points, number of centers, number of features, and standard deviation of the clusters were manipulated to provide a comprehensive set of testing scenarios. The synthetic datasets were designed to represent different challenges in clustering problems, including overlapping clusters, hierarchically nested clusters.

On the other hand, the real-world data were collected from Irvine Repository, publicly available source, in order to validate the effectiveness of the different clustering techniques in practical scenarios. These included datasets from the UCI Machine Learning Repository such as

1) “Iris” - one of the first datasets used for evaluation of classification methodologies.
2) “Wine” - dataset of chemical analysis of wines.
3) “Glass” - dataset of 6 types of glass; defined in terms of their oxide content.
4) “Landsat” - dataset of multi-spectral pixels values in 3x3 neighborhoods in a satellite image.
5) “Magic” - dataset of registration of high energy gamma particles in an atmospheric Cherenkov telescope.

These real-world datasets were chosen to provide a diverse range of data characteristics, including different numbers of dimensions, varying degrees of noise, and different distributions of data points. Before the clustering algorithms were applied, necessary preprocessing steps such as handling missing values, and normalization, and dimensionality reduction were carried out based on the specific requirements of each dataset.

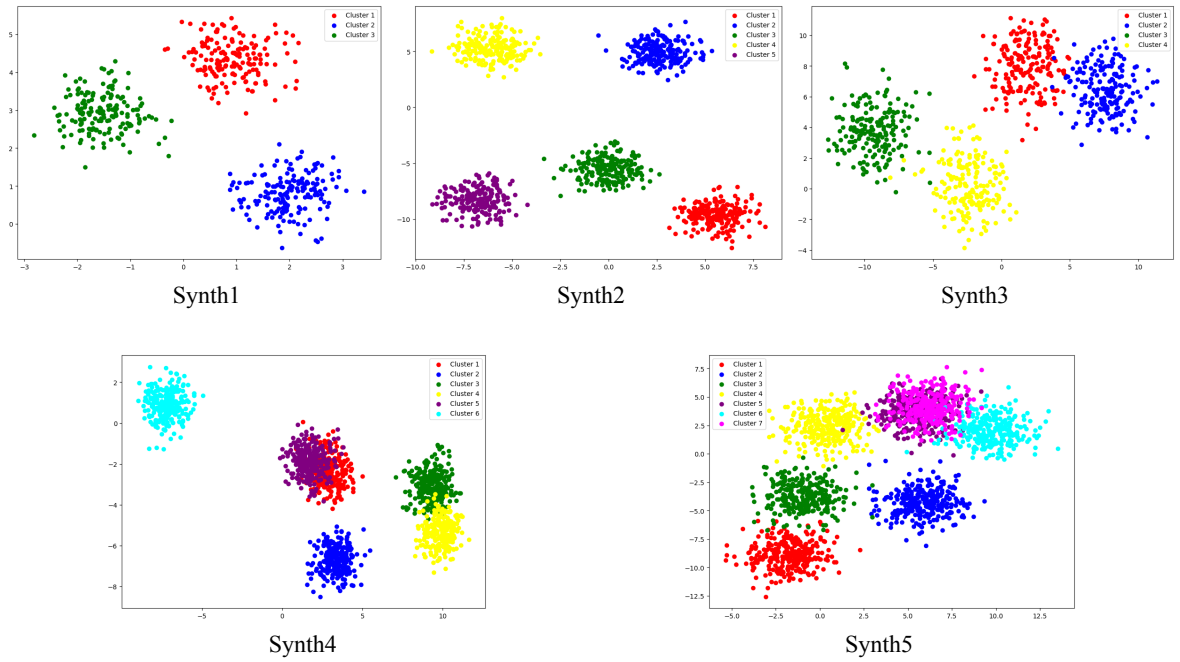

Diagram 0. 2-dimensional synthetic datasets

6. Methods

6.1 Elbow Algorithm

The Elbow method is a popular approach used in determining the optimal number of clusters in the K-means clustering algorithm. Its name is derived from the characteristic 'elbow' bend that appears in the plot of explained variance versus the number of clusters.

Here's how it works: the K-means algorithm is run multiple times, each time with an increasing number of clusters. For each iteration, the total within-cluster sum of squares (WCSS) [1] is calculated, which measures the compactness of the clustering, and hence the variance explained by the clusters. On the plot, at the point where adding another cluster doesn't significantly reduce the WCSS (the rate of decrease sharply shifts), an 'elbow' shape is formed in the plot. This 'elbow point' is considered as a good choice for the number of clusters.

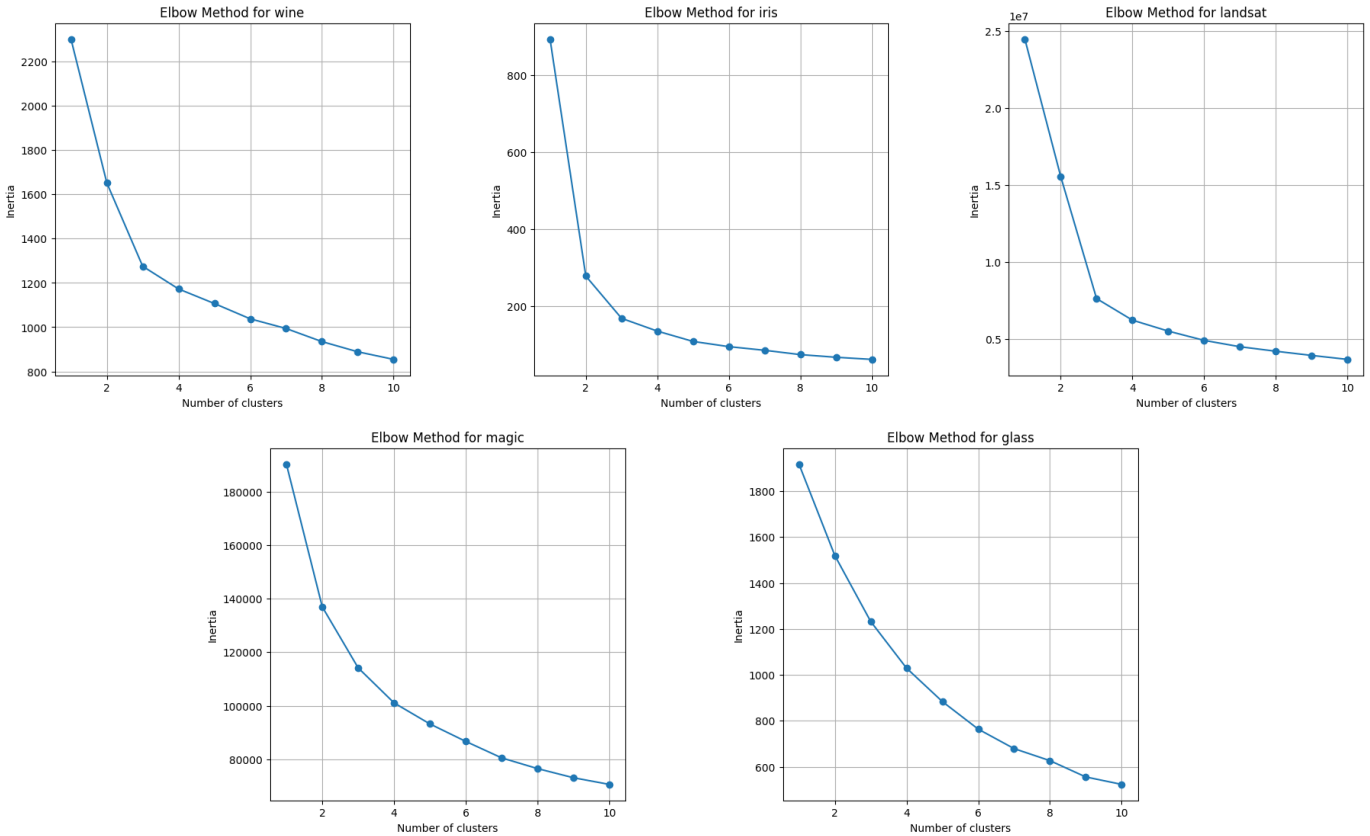

Diagram 1. elbow method graphs

6.2 Silhouette Algorithm

The Silhouette method is another approach used for determining the optimal number of clusters 'k' in the K-means algorithm. Unlike the Elbow method which relies on the total within-cluster variance, the Silhouette method leverages the concept of silhouette coefficients [2] to measure how close each data point in one cluster is to the data points in the neighboring clusters. This can provide a more direct measure of the quality of the clustering.

Here's how it works: The K-means algorithm is run multiple times, each time with a different number of clusters. For each iteration, the average Silhouette coefficient for all instances is calculated. This is then plotted against the number of clusters. The optimal number of clusters is the one that maximizes the average Silhouette coefficient.

6.3 Iterative Anomalous Clusters Algorithm (HSE)

The Iterative Anomalous Clusters (IAC) method is an advanced technique to determine the optimal number of clusters in a dataset. The primary objective of this algorithm is to identify clusters that stand out as different or "anomalous" with respect to the rest of the dataset.

Here's how it works: The algorithm starts with the entire dataset and identifies the farthest point from the origin [3]. This point serves as the centroid of the first anomalous cluster. The method then assigns points to this cluster if their distance [4] to the centroid is less than their distance to the origin. This process continues iteratively until the centroid of the anomalous cluster doesn't change between iterations. After identifying this anomalous cluster, the algorithm removes it from the dataset and repeats the process. This sequence-based analysis continues until no more anomalous clusters can be identified (i.e., when the size of the next anomalous cluster is smaller than a threshold t). The centroids of these anomalous clusters are then used as initial centroids for the K-means algorithm. The method returns the clusters identified by this final round of K-means as the final output.

6.4 Determine Cluster Number Without Clustering for Every K Based on Ratio of Variance to Range (DWCNK) Algorithm

The Determine Cluster Number Without Clustering for Every K Based on Ratio of Variance to Range (DWCNK - Determine Without Clustering Number (of clusters) K-means) method is an approach aimed at optimizing the K-means clustering process by reducing the number of iterations required to find an optimal number of clusters. Rather than running K-means clustering for every K (from 1 to a predefined maximum), the DWCNK method leverages the Ratio of Variance to Range (RVR) [5] and Dispersion-Width Ratio (DWR) [6] to identify an optimal number of clusters in a more efficient manner.

Here's how it works: The algorithm starts by performing a typical K-means clustering operation with a relatively large initial number of clusters (K0). The clusters are then ordered based on the mean Euclidean distances between the points within the same cluster, forming a sequence of clusters. A graph is then plotted with the y-axis representing the cumulative sum of the intra-cluster distances for each cluster in the sequence, and the x-axis representing the cluster sequence number. The optimal number of clusters, K_star, is determined as the x-coordinate of the point in the graph where the second derivative of the cumulative sum of intra-cluster distances is minimized.

6.5 The Gaussian Likelihood Score (GLS) Algorithm

The Gaussian Likelihood Score (GLS) is a method used to find the optimal number of clusters. This method takes advantage of the Gaussian distribution's properties to quantify how much the data points in each cluster follow a Gaussian distribution, and thus determine the best number of clusters.

Here's how it works: The method starts by iterating through all the clusters, and for each cluster, it measures the likelihood that the data points within that cluster follow a Gaussian distribution. This likelihood is computed based on the probability density function of the Gaussian distribution, given the mean and standard deviation of the data points within each cluster. The likelihoods of all the clusters are then summed up to obtain the total Gaussian likelihood score [7]. For the calculation of the Gaussian likelihood for each cluster the PCA is applied. The data points of each cluster are transformed into a single-dimensional space, and then the likelihood is computed based on the transformed data points. After obtaining the Gaussian likelihood scores for different numbers of clusters, the method computes the derivative of the scores and then the second derivative. The optimal number of clusters is identified as the number of clusters corresponding to the minimum of the second derivative [8].

6.6 Agglomerative Hierarchical Clustering Method

Hierarchical clustering is a method of cluster analysis that seeks to build a hierarchy of clusters. The Agglomerative Hierarchical Clustering (AHC) is a "bottom-up" approach where clustering starts with each element as a separate cluster and merges them into successively larger clusters.

Here's how it works: One way to determine the optimal number of clusters is to create a dendrogram, which is a tree-like diagram that records the sequences of merges or splits. In this approach, each merge is represented by a horizontal line. The y-coordinate of the horizontal line is the similarity of the two clusters that were merged, where cities are viewed as singleton clusters. By moving up from the bottom layer to the top node, a dendrogram allows us to reconstruct the history of merges that resulted in the depicted clustering. We can define the number of clusters by observing the dendrogram and identifying the largest vertical distance that doesn't intersect any of the other clusters.

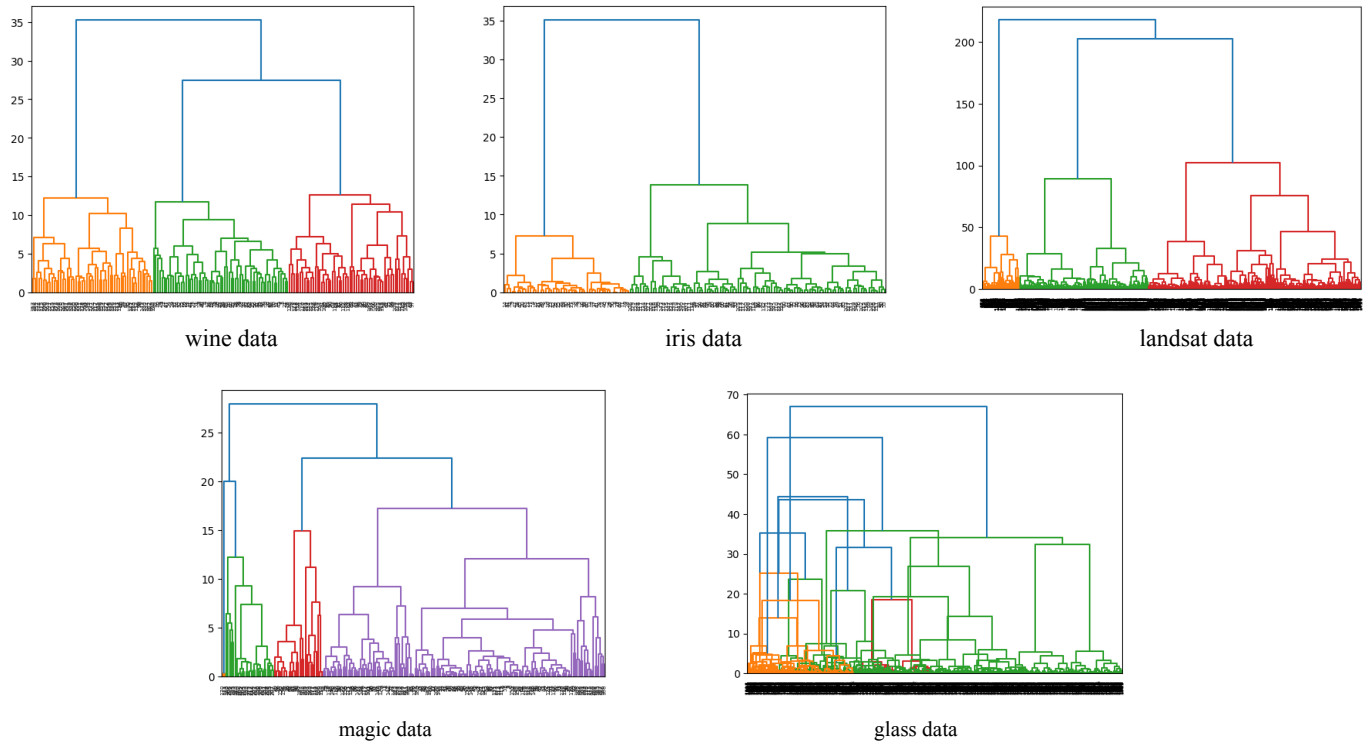

Diagram 2. Agglomerative clustering dendrograms

6.7 Data Depth Method

The Data Depth (DeD) method is an approach that uses the notion of 'depth' [9] in a multivariate data distribution to determine the optimal number of clusters. This concept of depth is based on the location of data points relative to the distribution of the dataset. For instance, the 'deeper' a point is, the closer it is to the center of the distribution.

Here's how it works: The DeD [12] score is calculated as the difference between the within-cluster depth (DW) and the between-cluster depth (DB). The DW [10] is essentially the mean absolute deviation from the median depth within the clusters, whereas the DB [11] is the mean absolute deviation from the median depth of the entire dataset, subtracted by DW. The optimal number of clusters is then selected as the one that maximizes the DeD score.

7. Experimental Results and Discussion

In the experimental phase of our project, we focused on evaluating the performance of our proposed novel methods for determining the optimal number of clusters for the K-Means algorithm and compared the results with the classical Elbow and Silhouette methods.

7.1 True K comparison

Datasets	Instances	Features	True K	hse	row	gaus	aggl	ded	elbow	sil
Wine	177	13	3	7	3	3	3	2	3	3
Iris	150	4	3	3	2	9	2	3	2	2
Landsat	1999	36	6	14	6	3	3	2	3	3
Magic	19019	10	2	6	13	3	2	4	2	2
Glass	213	9	2	8	4	5	4	4	3	2
Total			16	38	28	23	14	15	13	12
Sum of Errors				22	14	13	6	9	5	4

Diagram 3. Synthetic datasets true-k comparison

The table shows that Silhouette, Elbow, and Agglomerative Clustering methods had the smallest sum of errors, indicating their superior performance in determining the optimal number of clusters across these datasets. Meanwhile, Anomaly clusters (HSE), DWCNK (row), and Gaussian Likelihood methods had higher total errors, suggesting they may not perform as well with this particular set of data.

Datasets	Instances	Features	True K	hse	row	gaus	aggl	ded	elbow	sil
synth1	500	2	3	4	3	3	2	19	2	3
synth2	1000	2	5	6	5	5	3	20	2	5
synth3	750	2	4	4	4	4	2	20	2	3
synth4	1500	2	6	8	4	7	2	3	2	5
synth5	2000	2	7	5	6	8	2	2	2	6
Total			25	27	22	27	11	64	10	22
Sum of Errors				6	3	2	14	55	15	3

Diagram 4. Real datasets true-k comparison

The comparison indicates that the Silhouette method consistently performs well on both real and synthetic datasets, indicating their robustness. In contrast, methods like DeD and Agglomerative show varied performance across different types of datasets, suggesting their sensitivity to data characteristics.

7.2 Time comparison

Datasets	hse	row	gaus	aggl	ded	elbow	sil
Wine	1	1	1	1	1	1	1
Iris	1	1	1	1	1	1	1
Landsat	23	16	6	3	22	5	5
Magic	47	344	89	32	133	10	39
Glass	1	1	1	1	1	1	1
Total	73	363	98	38	158	17	47

Diagram 5. Real datasets time comparison

From the execution time comparison, the Elbow method shows the fastest computation time across the datasets. On the other hand, methods like DWCNK (row) and DeD require significantly more computation time, making them less suitable for large datasets. This indicates that while performance is essential, computational efficiency is also a critical factor to consider when selecting clustering methods.

Datasets	hse	row	gaus	aggl	ded	elbow	sil
synth1	1	2	1	1	2	1	1
synth2	2	3	1	1	3	1	1
synth3	2	3	1	1	2	1	1
synth4	4	5	1	2	5	1	2
synth5	3	7	1	3	7	1	3
Total	12	20	5	8	19	5	8

Diagram 6. Synthetic datasets time comparison

Analyzing execution times for synthetic datasets, we see that all methods performed faster compared to real datasets. Again, the Elbow method was the quickest, implying it handles variations in dataset size and complexity well. However, the ROW and DeD methods again took comparatively more time, suggesting these may not be ideal for larger or more complex synthetic datasets. Comparing both real and synthetic data, it's clear that computational times can vary significantly based on data type and structure.

7.3 Index comparison

The Davies-Bouldin index [13] is a metric for evaluating the quality of a clustering algorithm. It quantifies the average 'similarity' between clusters, where the similarity measure is a ratio of within-cluster distances to between-cluster distances. A lower value of the index indicates that clusters are compact (low intra-cluster distances) and well-separated (high inter-cluster distances). Hence, it is desirable to have a lower Davies-Bouldin index, which would signify better clustering.
In the research, seven different clustering methods were used and evaluated using the Davies-Bouldin index across five datasets. The Agglomerative clustering method gave the lowest Davies-Bouldin index (5.22), suggesting it provided the most effective clustering, with denser and better-separated clusters compared to the other methods, also elbow, silhouette and row have low indexes.

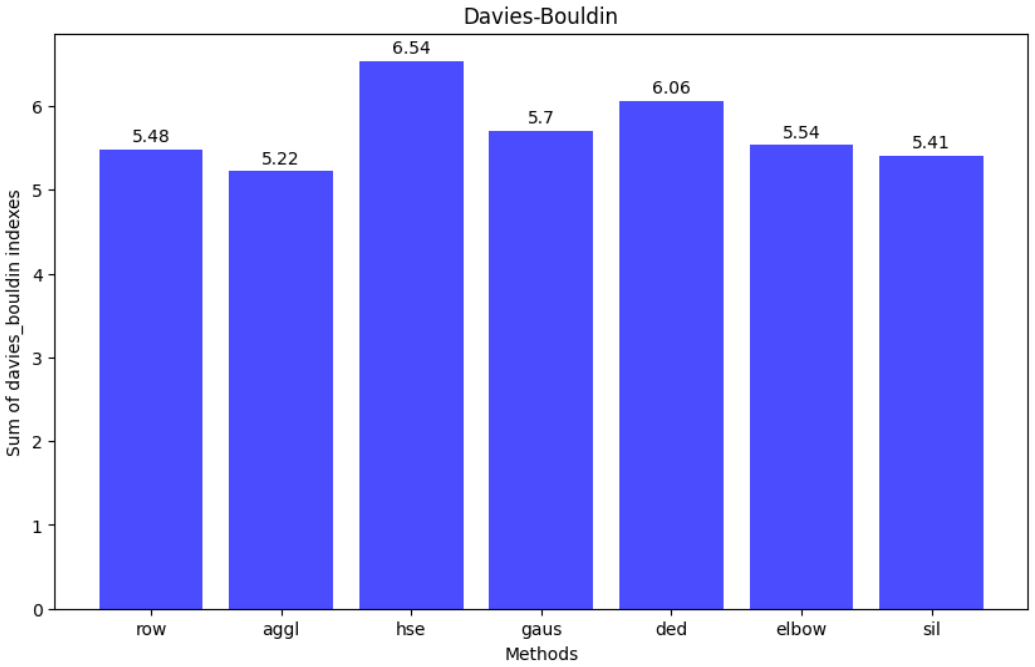

Diagram 7. Davies-Bouldin indexes

The Dunn index is a valuable metric for evaluating clustering algorithms. It measures the 'similarity' between clusters by comparing the smallest inter-cluster distances to the largest intra-cluster distances. Higher values of the Dunn index are desirable, as they suggest that clusters are compact and well separated.

In the research, seven distinct clustering techniques were assessed using the Dunn index across five datasets. It was observed that the 'sil' method demonstrated the highest Dunn index (0.75), suggesting that it offered the most effective clustering, producing dense and well-separated clusters. This signifies better performance compared to the other methods. Also row, agglomerative and elbow methods have high indexes.

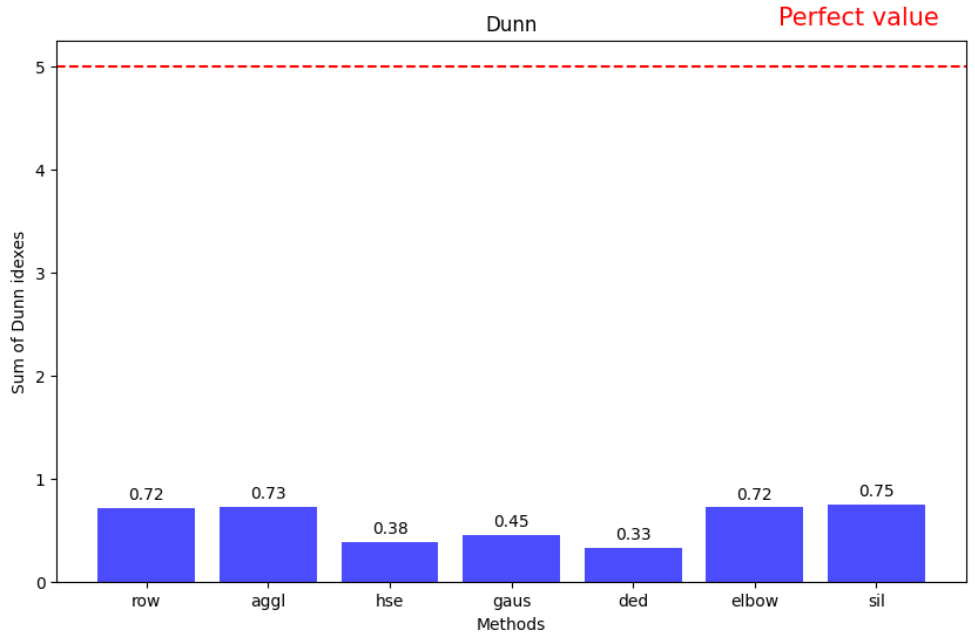

Diagram 8. Dunn indexes

The Silhouette index is a measure used to evaluate the quality of clustering. It calculates the average similarity of each instance to its own cluster compared to other clusters. The silhouette index ranges from -1 to 1, where a high value indicates that the object is well matched to its own cluster and poorly matched to neighboring clusters.

In my research, seven different clustering techniques were employed and evaluated using the silhouette index across five datasetsThe 'sil' method exhibited the highest silhouette index (2.05), indicating that it provided the most effective clustering, creating instances that were well-matched to their own clusters and poorly matched to others.

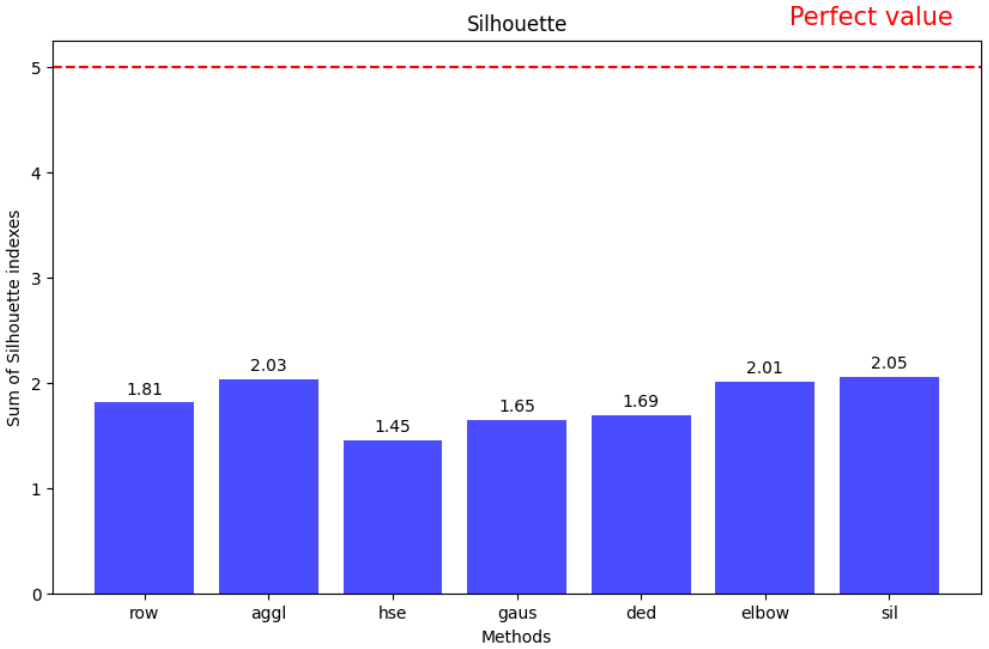

Diagram 9. Silhouette index

7.4 Pros and Cons of method

1. Gaussian Likelihood Method
Strengths:
- Can work well when the data are normally distributed.
- Can provide good results when clusters are spherical and have similar sizes.

Weaknesses:
- Performance degrades when the data are not normally distributed or when clusters are of significantly different sizes or shapes.

2. Agglomerative Clustering Method
Strengths:
- Can handle any kind of distance measure, making it more flexible for various types of data.
- Effective for identifying hierarchical relationships within data.

Weaknesses:
- May not perform well with large datasets due to its computational complexity.
- Difficulty in handling noisy data and outliers.

3. Anomalous Clustering Method (HSE)
Strengths:
- Can detect clusters with various shapes and sizes in data.
- May be robust to outliers and noise in data.

Weaknesses:
- May struggle with data containing high dimensional feature space.
- Can be computationally expensive for large datasets.

4. DeD Method
Strengths:
- Less sensitive to outliers as it uses the Mahalanobis distance, which takes into account the covariance of data.
- Can handle data with various shapes and sizes of clusters.

Weaknesses:
- Can be computationally expensive for large and high-dimensional datasets.
- Performance may degrade when the covariance matrix is ill-conditioned.

5. Silhouette Method
Strengths:
- Works well with data where clusters are clearly separable and cohesive.
- Can handle data with various shapes and sizes of clusters.

Weaknesses:
- Performance degrades when clusters are overlapping or when noise and outliers are present in the data.
- Computationally expensive for large datasets.

6. Elbow Method
Strengths:
- Works well when there's a clear 'elbow' or 'knee' in the distortion curve, often found in datasets with clear and distinct clusters.
- Relatively efficient even with large datasets.

Weaknesses:
- May not work well when the 'elbow' or 'knee' in the distortion curve is not clear or is ambiguous, which may happen in datasets with overlapping clusters or uniform data.

7. Ratio of Variance:
Strengths:
- Can work well for a wide range of data distributions, including those with non-spherical clusters.

Weaknesses:
- "Elbow point" might not be clearly defined for some datasets, leading to ambiguity in determining the optimal number of clusters.

Conclusion

The objective of this project was to investigate and contrast the performance of several modern methods of determining the optimal number of clusters against the conventional Elbow criterion within the K-means clustering algorithm context. This was conducted through a meticulous process of implementation and evaluation using both real-world and synthetically generated datasets.

Overall, it was observed that no single method performed uniformly superior across all the datasets. For instance, while the Silhouette method showed consistent results across both real and synthetic datasets, it was not always the best performer. The Elbow method, on the other hand, stood out for its computational efficiency, handling variations in dataset size and complexity well.

However, methods such as the Determine Cluster Number Without Clustering for Every K Based on Ratio of Variance to Range (DWCNK) and Data Depth (DeD) method, despite their promising concepts, were found to require significantly more computation time, making them less suitable for larger datasets.

The project also highlighted the importance of considering the characteristics of the data at hand when selecting the most suitable method for determining the optimal number of clusters. Different methods were observed to have varied performances across different types of datasets, suggesting their sensitivity to data characteristics.

Additionally, the quality of clustering, as evaluated by metrics like the Davies-Bouldin index, Dunn index, and Silhouette index, varied for different methods. For instance, while the Agglomerative clustering method yielded the most effective clustering as per the Davies-Bouldin index, the Silhouette method stood out when evaluated using the Dunn and Silhouette indexes.

To sum up, this project underscores the diverse capabilities and limitations of various contemporary and traditional techniques in establishing the ideal cluster count within the context of K-means clustering. This deepens our understanding of their real-world applicability and constraints. Crucially, the results highlight a potent necessity for continued exploration and innovation in this field, specifically in formulating techniques that can effectively and precisely identify the ideal number of clusters across a broad spectrum of datasets. This, in turn, can augment the resilience and effectiveness of clustering solutions for tasks reliant on data analytics.

Formulas

[1] The K-means objective function or the within-cluster sum of squares (WCSS) for Elbow method

$$WCSS = \sum_{i=1}^{k} \sum_{x \in C_i} ||x - \mu_i||^2$$

Where k - is the number of clusters, C_i - represents the i^{th} cluster, x - is a point within cluster C_i, μ_i - is the centroid of the i^{th} cluster.

[2] Silhouette coefficient for the Silhouette method to choose k

$$s(x) = \frac{b(x) - a(x)}{\max\{a(x), b(x)\}}$$

Where $a(x)$ is the mean intra-cluster distance, the average distance between the instance x and all other points in the same cluster. $b(x)$ is the mean nearest-cluster distance, the minimum average distance from x to instances in all other clusters.

[3] The farthest point 'c' from a reference point 'origin' for IAC method

$$c = \arg\max_{x \in X} D(x, \text{origin})$$

[4] The distance between points ‘p1’ and ‘p2’ in a d-dimensional space for IAC method

$$D(p1, p2) = \sqrt{\sum_{i=1}^{d} (p1_i - p2_i)^2}$$

[5] Mathematical computation of RVR for DWCNK method

$$RVR = \frac{1}{k} \sum_{i=1}^{k} \frac{Var_i}{Range_i}$$

In the formula above, k is the number of clusters, Var_i is the variance of the i-th cluster, and $Range_i$ is the range of the i-th cluster.

[6] Mathematical computation of DWR for DWCNK method

$$DWR = \frac{1}{k} \sum_{i=1}^{k} \frac{Dispersion_i}{Width_i}$$

In the formula above, k is the number of clusters, $Dispersion_i$ is the dispersion (usually measured as the standard deviation) of the i-th cluster, and $Width_i$ is the width (or range) of the i-th cluster.

[7] Compute the Gaussian likelihood score for each cluster

$$P = \sum_{j=1}^{n} \log(\text{normpdf}(x, \mu_j, \sigma_j))$$

Where 'normpdf' represents the Gaussian probability density function, 'x' represents the data points, 'μ_j', is the mean, 'σ_j' is the standard deviation, and 'n' is the number of clusters.

[8] Identify the optimal number of clusters for GLS method

$$k^* = \arg\min_i dd_i$$

- $d = P_i - P_{i-1}$
- $dd = \frac{d_i}{d_{i-1}}$

[9] Depth of a point

$$D(x) = \frac{1}{N} \sum_{i=1}^{N} |x_i - x|$$

[10] Within-cluster depth

$$DW = \frac{1}{k} \sum_{j=1}^{k} \left(\frac{1}{|C_j|} \sum_{x \in C_j} |D(x, C_j) - median(D(X, C_j))| \right)$$

[11] Between-cluster depth

$$DB = \frac{1}{N} \sum_{i=1}^{N} |D(x_i, X) - median(D(X, X))| - DW$$

[12] Data depth score

$$DeD = DW - DB$$

[13] Davies-Bouldin index

$$DBI = \frac{1}{n} \sum_{i=1}^{n} \max_{i \neq j} \left(\frac{s_i + s_j}{d_{ij}} \right)$$

[14] Dunn index

$$DI = \min_{1 \leq i \leq n} \left(\min_{i \neq j} \left(\frac{d(i,j)}{\max_k d'(k)} \right) \right)$$

[15] Silhouette index

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}$$

References

Algorithms:

1. Elbow
   https://www.geeksforgeeks.org/elbow-method-for-optimal-value-of-k-in-kmeans/

2. Silhouette
   https://scikit-learn.org/stable/auto_examples/cluster/plot_kmeans_silhouette_analysis.html

3. Iterative Anomalous Clusters (HSE)
   https://www.hse.ru/ba/ami/students/diplomas/470537628

   Code
   https://colab.research.google.com/drive/1fecsm8hY55PwPxBOxVCure-F2phNkt7D#scrollTo=aYCzO7MKj_ZB

4. DWCNK
   https://www.hindawi.com/journals/mpe/2022/6866747/

5. Gaussian Likelihood Score
   https://medium.com/@mchenebaux/choosing-the-right-number-of-clusters-using-the-gaussian-likelihood-score-30bce5ad6eac

   Code
   https://github.com/kerighan/gls

6. Agglomerative
   https://www.youtube.com/watch?v=4DInt3H2UNE

7. Data Depth
   https://link.springer.com/article/10.1007/s41019-019-0091-y

Datasets:

1. Iris
   https://archive.ics.uci.edu/dataset/53/iris

2. Wine
   https://archive.ics.uci.edu/dataset/109/wine

3. Landsat
   https://archive.ics.uci.edu/dataset/146/statlog+landsat+satellite

4. Magic
   https://archive.ics.uci.edu/dataset/159/magic+gamma+telescope

5. Glass
   https://archive.ics.uci.edu/dataset/42/glass+identification

6. Synthetic data
   sklearn.datasets.make_blobs — scikit-learn 1.3.0 documentation

Comparison:

1. Dunn index
   https://www.geeksforgeeks.org/dunn-index-and-db-index-cluster-validity-indices-set-1/

2. Davies-Bouldin index
   https://scikit-learn.org/stable/modules/generated/sklearn.metrics.davies_bouldin_score.html

3. Silhouette index
   https://scikit-learn.org/stable/auto_examples/cluster/plot_kmeans_silhouette_analysis.html