International Journal of Multidisciplinary Approach Research and Science
E-ISSN 2987-226X P-ISSN 2988-0076
Volume 2 Issue 01, January 2024, Pp. 385-396
DOI: https://doi.org/10.59653/ijmars.v2i01.504

Evaluation of Data Clustering Accuracy using K-Means Algorithm

Suraya[1], Muhammad Sholeh[2*], Uning Lestari[3]
Computer Systems Engineering Study Programme, Faculty of Applied Science, AKPRIND Institute of Science & Technology Yogyakarta, Indonesia | suraya@akprind.ac.id[1]
Informatics Study Programme, Faculty of Information Technology and Business, AKPRIND Institute of Science & Technology Yogyakarta, Indonesia | muhash@akprind.ac.id[2]
Informatics Study Programme, Faculty of Information Technology and Business, AKPRIND Institute of Science & Technology Yogyakarta, Indonesia | ules@akprind.ac.id[3]
Correspondence Author*

Received: 05-12-2023 *Reviewed: 10-12-2023* *Accepted: 21-12-2023*

Abstract

Data clustering is one of the methods in data science that is often used in data analysis. This method is used in making groupings from a collection of datasheets. Data clustering is done to find patterns or relationships between data. This research aims to evaluate the accuracy of data clustering using K-Means algorithm on wine datasheet. Wine datasheet has 13 features that describe the chemical characteristics of three types of wine. The clustering process must produce the best clustering evaluation metrics. The evaluation metric is done through comparison between the clustering results of K-Means algorithm with Davies Bouldin and Silhouette. The research steps involved data standardization, selection of the optimal number of clusters, and assessment of clustering accuracy. The research method uses KDD which consists of pre-processing, transformation, model building and model evaluation. Experimental results show that appropriate parameters and cluster initialization can improve clustering evaluation metrics. The clustering results show that the normalized datasheet produces evaluation metrics for Davies Bouldin 2 groups and Silhouette produces 3 groups. Before normalization, Davies Bouldin results in 7 groups and Silhouette results in 2 groups. In conclusion, this study produced different evaluation metrics between normalized and non-normalized datasheets. The selection of the number of groups chosen depends on the context of the data analysis performed and is selected into 3 groups which can be labelled "Superior Variety", the second group "Intermediate Variety" and the third group "Standard Variety".

Keywords: Metrics, evaluation, normalized clustering, labels

Introduction

In the rapidly growing digital era, data processing and analysis have become critical elements in generating valuable information for various fields. One popular approach in data analysis is clustering, which aims to identify patterns or structures hidden in datasets.

Clustering is a data analysis method that aims to group objects into groups or clusters based on the similarity of certain characteristics. In this context, objects that have similarities will be placed in one group, while objects that are different will be placed in different groups. The basic concepts in clustering involve the similarity between objects, the formation of clusters as a result of the clustering process, the centroid as the group centre point in the K-Means algorithm, and the measurement of the distance between objects in the feature space (Cielen et al., 2016), (Ozdemir, 2017).

There are various clustering methods that can be applied, the selection of methods is adjusted to the characteristics of the data and the purpose of the analysis. Algorithms used in clustering models include K-Means, Hierarchical Clustering, DBSCAN, and Gaussian Mixture Models (Deny Jollyta, Muhammad Siddik, Herman Mawengkang, 2021), (Mathur, 2019). The clustering process can be used for data structure understanding, customer segmentation in the marketing industry, image analysis, and other analysis purposes. The selection of an appropriate clustering method and understanding of the results is critical to ensure the relevance and success of the analysis (Amanda & Veronica Sitorus, 2021), (Garang, 2022).

Literature Review

The K-Means algorithm is one of the popular data analysis techniques that is widely used in the data clustering process (Purba et al., 2022). It uses a partitioning system to group data based on their similarity, with each cluster represented by a centroid point (Informatics & Polinema, 2020). The process involves iteratively updating the centroid points until the data points are optimally grouped into clusters (Asmiatun et al., 2019). Measuring the accuracy of the K-Means algorithm is very important in evaluating the effectiveness of the clustering process (Awaludin, 2014). The accuracy of the algorithm is influenced by several factors including the initial centroid point value, the number of clusters, and the distance metric used to calculate the similarity between data points (Faizah et al., 2020), (Dewi & Pramita, 2019).

The K-Means algorithm has been applied in various studies to cluster data and evaluate the accuracy of the clustering process (Kurniadi et al., 2023), (Tambunan, 2021), (Listiani et al., 2019).Research conducted by Nurjanah (Nurjanah & Arifin, 2021), applied the K-Means method in analysing travel review data. K-Means, as a clustering algorithm, can help identify patterns and groups of words in reviews, enabling a deeper understanding of users' impressions of a place. Natsir (Dewi & Pramita, 2019), made clustering on book data borrowed in the library, Mujiono (Muliono & Sembiring, 2019), used the Kmeans algorithm for clustering data on the tri darma activities of lecturers and Priyatman, H (Priyatman et al., 2019), made a clustering model for use in promotional mapping.

The results of the model development must be seen the resulting accuracy value. The process of evaluating the accuracy of the clustering model is used to calculate the best clustering value. Methods for measuring the accuracy of clustering models can include using such as Silhouette Score, Davies-Bouldin Index and Adjusted Rand Index (ARI).

Evaluation of clustering models requires the selection of metrics that are appropriate to the task context and data characteristics. These metrics are often used together to provide a more comprehensive picture of the quality of clustering produced by the model. By understanding the strengths and weaknesses of each metric, researchers and practitioners can make more informed decisions in assessing the performance of a clustering model and understand the extent to which the model is able to uncover meaningful structure in the data in the absence of true class labels.

The Davies-Bouldin index is calculated with respect to two main aspects, namely the cohesiveness and unity of each group. Cohesiveness measures the extent to which points within a group are similar among themselves, while unity measures the extent to which one group is distinct from another. A Davies-Bouldin index value is calculated for each group by comparing the cohesiveness and unity with other groups. This process results in a series of Davies-Bouldin index values, where lower values indicate better clustering results. Furthermore, these values can be averaged to get an overall picture of the clustering accuracy of the entire dataset (Jollyta et al., 2019), (Sholeh & Aeni, 2023), (Quinthara et al., 2023).

The clustering accuracy evaluation process using the Silhouette method measures how well the division of groups is able to separate between groups and the extent to which members in one group are similar to each other. The Silhouette Score is calculated by comparing the average distance of a point to a point in the same group (a) with the closest average distance between that point and a point from another group (b). The Silhouette Score for each point is then obtained from the formula (b - a) / max (a, b). The Silhouette value ranges from -1 to 1, where a positive value indicates that the point fits better into its group compared to other groups (Orisa, 2022), (Vania & Sari, 2023), (Paembonan & Abduh, 2021).

In this study, the data clustering process was carried out using the K-Means algorithm and using accuracy with the Davis Bouldin and Silhouette methods. The purpose of data clustering includes that existing data can be grouped into several groups, which in one group is a collection of data that has similar similarities. Clustering enables identification and a deeper understanding of different grape varieties, chemical composition, and other attributes. This process can assist in grouping wines based on similar characteristics, facilitating data analysis and interpretation. In addition, clustering can be directed towards the determination of wine quality, distinguishing between high and low quality wines.

Research Method

Datasheet

The research was conducted using a datasheet that was processed using a public datasheet, namely a wine datasheet. This datasheet was obtained from

https://archive.ics.uci.edu/dataset/109/wine. The datasheet consists of 13 features as much as 178 data.

Research Stages

In data mining method research using the Knowledge Stages method in KDD begins with the selection of datasets that match the purpose of the analysis. The next steps in the KDD method are pre-processing, transformation, model building and model evaluation. The pre-processing stage is done by data cleaning, handling missing values, and at the data transformation stage to fit the needs of the analysis. This stage ensures data quality and integrity before moving on to the next stage. The modelling stage in this research uses a clustering model with the K-Means algorithm which is used to group data based on the similarity of its characteristics. The last stage is to see the accuracy value of the model created. The accuracy method uses Davies Bouldin and Silhouette. The KKD method used in the research process is presented in Figure 1.

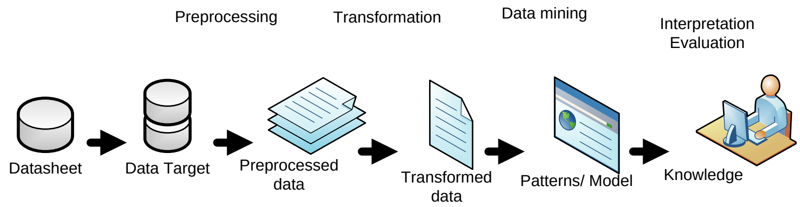


Figure 1, Research stages with the KDD method

K-Means Algorithm

K-Means algorithm is a clustering method used to group data points into similar groups. The steps of the K-Means algorithm are:

1. Centroids initialization:
   The first step is to determine how many clusters (k) you want to create and randomly select the centroid points. This centroid value is the starting point of clustering.
2. Data Clustering:
   Performs the process of assigning each data point to the group that has the closest centre. Group centres are measured using a distance metric, often the Euclidean distance.

$$d(P,Q) = \sqrt{p1 - q1)^2 + (p2 - q2)^2 + \cdots + (pn - qn)^2} \quad \ldots\ldots\ldots\ldots\ldots\ldots (1)$$

3. Update of group centroids:
   Calculate the new centroids by calculating the average of each group generated in the process of ledge 2. The process of calculating the new centroids is done using the data that has been attributed to the group.

$$uj\ (t+1) = \frac{1}{Nsj} \sum_{j \in sj} xj \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2)$$

$uj\ (t+1)$: the new centroid at iteration (t+1), Nsj : the number of data in cluster sj.

4. Looping process
   Repeat steps 2 and 3 until there is no significant change in cluster attribution or until the specified number of iterations.
5. Convergence Evaluation:
   Convergence evaluation can be done by monitoring the change in cluster centres or by checking if the cluster attribution does not change within a few iterations.
6. Termination:
   The algorithm stops when the stopping criteria are met, such as reaching the maximum number of iterations or when there is no significant change in the group attribution.

Results and Discussion

The pre-processing process involves examining the entire data to identify missing values, outliers or other potential problems that may affect the quality of the analysis. As a result of the identification of dirty data, the process of cleaning the data identified can interfere with the modelling process. The stages of pre-processing are shown in Table 1.

Table 1 Stages in pre-processing

No	Command	Stages	Result
1	Checking for empty data	df.isnull().sum()	No empty data found
2	Checking data type	df.info()	All data types are numeric
3	Removing duplicate data	df=df.drop_duplicates()	Same data has been deleted
4	Checking data consistency	for column in df.columns: unique_values = df[column].unique() print(f"Unique Value in Value Column '{column}':") print(unique_values) print("\n")	All data is consistent with the data entry
5	Checking outlier data	plt.figure(figsize=(39, 6)) sns.boxplot(data=df) plt.title("Boxplot untuk Semua Fitur") plt.show()	There is no outlier data

Transformation Stages

The transformation process can be done with data normalization. The data normalization process is done by scaling the variables on the datasheet to have the same range. Datasheets that have not been normalized are shown in Figure 2.

	Alcohol	Malic_Acid	Ash	Ash_Alcanity	Magnesium	Total_Phenols	Flavanoids	Nonflavanoid_Phenols	Proanthocyanins
0	14.23	1.71	2.43	15.6	127	2.80	3.06	0.28	2.29
1	13.20	1.78	2.14	11.2	100	2.65	2.76	0.26	1.28
2	13.16	2.36	2.67	18.6	101	2.80	3.24	0.30	2.81
3	14.37	1.95	2.50	16.8	113	3.85	3.49	0.24	2.18
4	13.24	2.59	2.87	21.0	118	2.80	2.69	0.39	1.82

Figure 2 Wine datasheet that has not been normalized

The transformation process is performed using the StandardScaler() library. This function is applied to attribute values to transform the distribution of values to have a mean (μ) around 0 and a standard deviation (σ) around 1. StandardScaler follows the following mathematical formula:

$$z = \frac{x-\mu}{\sigma} \quad \text{..... .3}$$

where:

z = the transformed value (the scale has been changed),

x = the original value of the attribute,

μ = the mean of the attribute,

σ = is the standard deviation of the attribute.

The results of the normalization process are in Figure 3

	Alcohol	Malic_Acid	Ash	Ash_Alcanity	Magnesium	Total_Phenols	Flavanoids	Nonflavanoid_Phenols	Proanthoc
0	1.518613	-0.562250	0.232053	-1.169593	1.913905	0.808997	1.034819	-0.659563	1.
1	0.246290	-0.499413	-0.827996	-2.490847	0.018145	0.568648	0.733629	-0.820719	-0.
2	0.196879	0.021231	1.109334	-0.268738	0.088358	0.808997	1.215533	-0.498407	2.
3	1.691550	-0.346811	0.487926	-0.809251	0.930918	2.491446	1.466525	-0.981875	1.
4	0.295700	0.227694	1.840403	0.451946	1.281985	0.808997	0.663351	0.226796	0.

Figure 3 Wine datasheet that has been normalized

Stages of model building

The process of creating a clustering model using the K-means algorithm by comparing the original datasheet with the normalized datasheet. After the clustering model is applied, evaluation is carried out using the Davies-Bouldin index and Silhouette Score metrics. This evaluation is used to ensure adequate clustering results. Analysis of the results is done to understand the patterns and characteristics of the clusters formed, with interpretation taking into account attributes that may distinguish the clusters.

Clustering with the Davies-Bouldin evaluation metric

The Davies-Bouldin index can be used to give an idea of how well each group is separated from each other and how close the groups are. Lower index values indicate that clustering is better, where each group has its own centre and the groups are well separated. Conversely, higher index values may indicate overlap or fuzziness in the clustering.

The clustering process with the datasheet before normalization is shown in Figure 4 and the clustering process with the datasheet after normalization is shown in Figure 5. Tests were conducted with clustering ranging from k=2 to k=10.

```
results = {}
for i in range(2,11):
    kmeans = KMeans(n_clusters=i, random_state=30)
    labels = kmeans.fit_predict(df)
    db_index = davies_bouldin_score(df, labels)
    print("K = ",i,' = ', db_index)
    results.update({i: db_index})
```

```
K =  2  =  0.47878357591419884
K =  3  =  0.5342431775436268
K =  4  =  0.5443446908806354
K =  5  =  0.5744186247681738
K =  6  =  0.5224150770101008
K =  7  =  0.4578713662464749
K =  8  =  0.4974002441025693
K =  9  =  0.5307163453404443
K =  10  =  0.5356298301454323
```

```
plt.plot(list(results.keys()), list(results.values()))
plt.xlabel("Number of clusters")
plt.ylabel("Davies-Boulding Index")
plt.show()
```

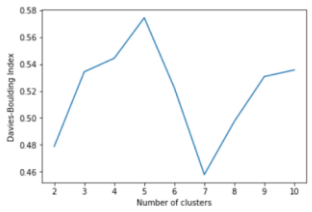


Figure 4 Clustering results with Davies-Bouldin evaluation metric before normalization

```
results = {}
for i in range(2,11):
    kmeans = KMeans(n_clusters=i, random_state=30)
    labels = kmeans.fit_predict(data_scaled )
    db_index = davies_bouldin_score(data_scaled , labels)
    print("K = ",i,' = ', db_index)
    results.update({i: db_index})
```

```
K =  2  =  1.5259965836052092
K =  3  =  1.389187977718165
K =  4  =  1.7007942237746472
K =  5  =  1.670308848848912
K =  6  =  1.62606294967109
K =  7  =  1.8260458987562331
K =  8  =  1.5399166070898764
K =  9  =  1.7351257146786114
K =  10  =  1.7309244626292042
```

```
plt.plot(list(results.keys()), list(results.values()))
plt.xlabel("Number of clusters")
plt.ylabel("Davies-Boulding Index")
plt.show()
```

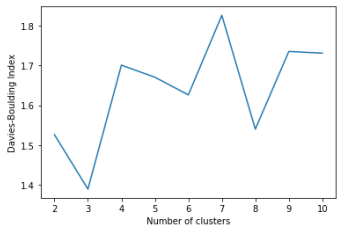


Figure 5 Clustering results with Davies-Bouldin evaluation metric after normalization

Clustering with Silhouette Score evaluation metric

The Silhouette Score evaluation metric is used to evaluate the quality of clustering by measuring how well each data in a group compares with other groups. The Silhouette Score ranges from -1 to 1, where higher values indicate better clustering. The clustering process with the datasheet before normalization is shown in Figure 6 and the clustering process with the datasheet after normalization is shown in Figure 7. Clustering tests were carried out with clustering processes ranging from k=2 to k=10.

```
SK = range(2,11)
sil_score = []
for i in SK:
    labels=cluster.KMeans(n_clusters=i,init="k-means++",
           random_state=200).fit(df).labels_
           score = metrics.silhouette_score(df,labels,metric="euclidean"
          ,sample_size=1000,random_state=200)
    sil_score.append(score)
    print ("Silhouette score for k(clusters) = "+str(i)+" is "+
       str(metrics.silhouette_score(df,labels,
       metric="euclidean",sample_size=1000,random_state=200)))
```

```
Silhouette score for k(clusters) = 2 is 0.6568536504294319
Silhouette score for k(clusters) = 3 is 0.5711381937868844
Silhouette score for k(clusters) = 4 is 0.5620915308951067
Silhouette score for k(clusters) = 5 is 0.5489993239795691
Silhouette score for k(clusters) = 6 is 0.5656413100386389
Silhouette score for k(clusters) = 7 is 0.5616196384056904
Silhouette score for k(clusters) = 8 is 0.5486213604076143
Silhouette score for k(clusters) = 9 is 0.5279990578758652
Silhouette score for k(clusters) = 10 is 0.5160641215582014
```

```
sns.lineplot(x = 'Clusters', y = 'Sil Score', data = sil_centers, marker="+")
```

```
<AxesSubplot:xlabel='Clusters', ylabel='Sil Score'>
```

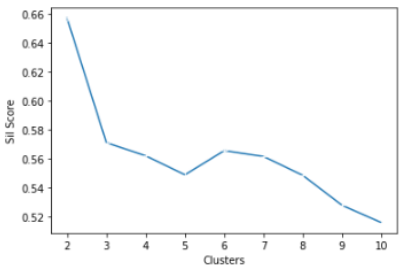
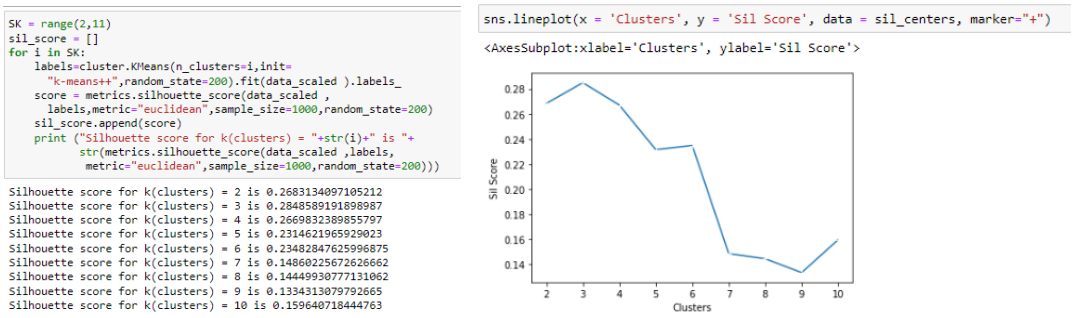


Figure 6 Clustering results with Silhouette Score evaluation metric before normalization



```
SK = range(2,11)
sil_score = []
for i in SK:
    labels=cluster.KMeans(n_clusters=i,init=
      "k-means++",random_state=200).fit(data_scaled ).labels_
    score = metrics.silhouette_score(data_scaled ,
      labels,metric="euclidean",sample_size=1000,random_state=200)
    sil_score.append(score)
    print ("Silhouette score for k(clusters) = "+str(i)+" is "+
           str(metrics.silhouette_score(data_scaled ,labels,
            metric="euclidean",sample_size=1000,random_state=200)))
```

```
Silhouette score for k(clusters) = 2 is 0.2683134097105212
Silhouette score for k(clusters) = 3 is 0.2848589191898987
Silhouette score for k(clusters) = 4 is 0.2669832389855797
Silhouette score for k(clusters) = 5 is 0.2314621965929023
Silhouette score for k(clusters) = 6 is 0.23482847625996875
Silhouette score for k(clusters) = 7 is 0.14860225672626662
Silhouette score for k(clusters) = 8 is 0.14449930777131062
Silhouette score for k(clusters) = 9 is 0.1334313079792665
Silhouette score for k(clusters) = 10 is 0.159640718444763
```

```
sns.lineplot(x = 'Clusters', y = 'Sil Score', data = sil_centers, marker="+")
```

```
<AxesSubplot:xlabel='Clusters', ylabel='Sil Score'>
```


Figure 7 Clustering results with Silhouette Score evaluation metric after normalization.

Model evaluation stage

Evaluation of clustering models can be done by measuring the quality and effectiveness of data clustering. The evaluation metrics used are Silhouette Score and Davies-Bouldin Index. The results of each clustering obtained are analyzed to see the best results from each evaluation metric. Analysis and interpretation of the results becomes a process to select or determine the value of the evaluated metrics of the resulting clustering. A comparison of the results of the evaluation metrics for each method is presented in Table 2.

Table 2 Evaluation metric results

Many clusters	Datasheet before normalisation		Datasheet after normalisation	
	Davies Bouldin	Silhouette	Davies Bouldin	Silhouette
2	0.47	0.656	1.525	0.268
3	0.53	0.571	1.389	0.284
4	0.54	0.562	1.707	0.266
5	0.574	0.548	1.670	0.231
6	0.522	0.565	1.626	0.234
7	0.457	0.561	1.826	0.148
8	0.497	0.548	1.539	0.144
9	0.530	0.527	1.735	0.133
10	0.535	0.516	1.730	0.159

Based on table 1, the clustering process is carried out from k = 2 to k = 10 and the evaluation results obtained, the datasheet that has not been normalized for the Davies Bouldin evaluation metric is the best clustering of 7 with a value of 0.457 and for the Silhouette evaluation metric the best clustering is 2 with a value of 0.656. The results of data clustering can be displayed in the form of 2 dimensions or 3 dimensions. The visualization results in the form of 2 dimensions are in Figure 8 and visualization in the form of 3 dimensions are in Figure 9.

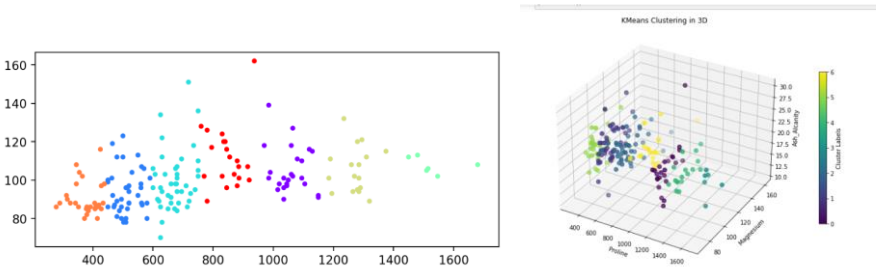


Figure 8. 2-Dimensional and 3-Dimensional Visualization of Davies Bouldin evaluation metrics Before normalization

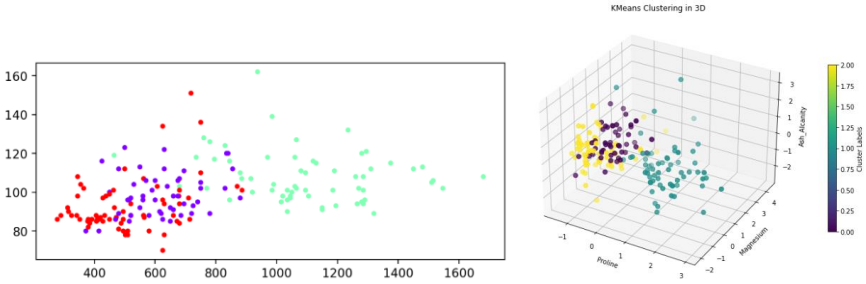


Figure 9 Visualization of 2-Dimensional and 3-Dimensional Silhouette evaluation metrics Before normalization.

After normalization, the clustering results produce different evaluation metric values. The results of the Davies Bouldin evaluation metric are the best grouping of 3 with a value of 1.389 and for the Silhouette evaluation metric the best grouping of 3 with a value of 0.284. The visualization results of the recommended clustering are shown in Figures xx and yy. Figure 10 displays the visualization in 2 dimensions for both Davies Bouldin and Silhouette Figure 11 displays the visualization in 3 dimensions for both Davies Bouldin and Silhouette.

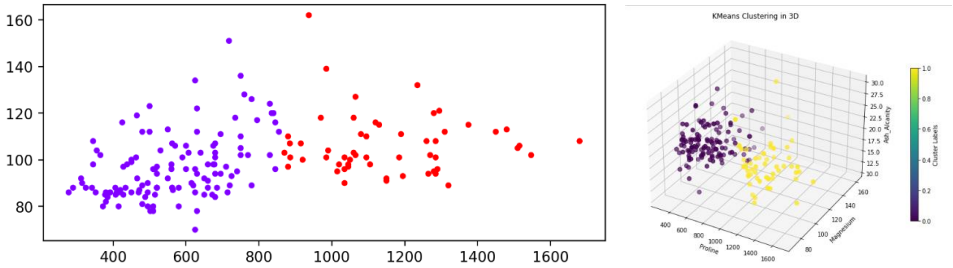


Figure 10 2-Dimensional and 3-Dimensional visualisation of Davies Bouldin evaluation metrics after normalisation

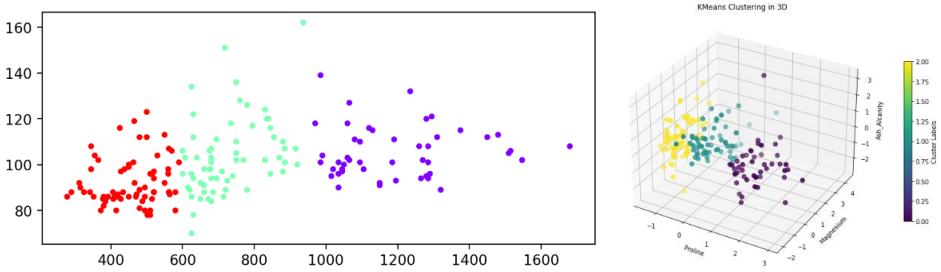


Figure 11 Visualization of 2-Dimensional and 3-Dimensional Silhouette evaluation metrics after normalization.

Based on the results of the evaluation metrics, especially the Silhouette evaluation results, many clusters of 3 groups were determined. The process of labelling the clustering results is done by understanding the structure and characteristics of the groups generated by the clustering model. In labelling the wine datasheet that is clustered into three, the first group can be labelled "Superior Variety", the second group "Intermediate Variety" and the third group "Standard Variety".

Conclusion

The results of clustering analysis on wine datasheets using datasheets that have not been normalized and with the normalization process produced different clustering evaluation metrics. Based on the Davies Bouldin and Silhouette evaluation metrics, the clustering results were selected from the three clustering results generated by the Silhouette evaluation metric and the normalized data. The results of the three groupings were labelled with the names "Superior Varieties," "Intermediate Varieties," and "Standard Varieties," respectively. This labelling is expected to provide an easier and more understandable interpretation of each group.

References

Amanda, & Veronica Sitorus, M. (2021). Penerapan Algoritma K-Means Clustering Untuk Pengelompokan Konsumsi Produk Kosmetik milik PT Cedefindo. *Jurnal Ilmiah MIKA AMIK Al Muslim*, *V*(2), 63–68.

Asmiatun, S., Wakhidah, N., Putri, A. N., & Kunci, K. (2019). *Identifikasi Kondisi Permukaan Jalan Menggunakan K-Means Clustering Road Surface Conditions Identification Using K-Means Clustering*. *November 2019*, 23–30.

Awaludin, M. (2014). Penerapan Algoritma K-Means Clustering Pada K-Harmonic Means Untuk Schedule Preventive Maintenance Service. *Jurnal Sistem Informasi Universitas Suryadarma*, *6*(1), 1–17. https://doi.org/10.35968/jsi.v6i1.271

Cielen, D., Meysman, A. D. B., & Ali, M. (2016). *Introducing Data Science: Big Data, Machine Learning, and more, using Python tools - PDFDrive.com*. Manning Publications.

Deny Jollyta , Muhammad Siddik , Herman Mawengkang, S. E. (2021). *Teknik Evaluasi Cluster Solusi Menggunakan Python Dan Rapidminer*. Deepublish Publisher.

Dewi, D. A. I. C., & Pramita, D. A. K. (2019). Analisis Perbandingan Metode Elbow dan Silhouette pada Algoritma Clustering K-Medoids dalam Pengelompokan Produksi Kerajinan Bali. *Matrix : Jurnal Manajemen Teknologi Dan Informatika*, *9*(3), 102–109. https://doi.org/10.31940/matrix.v9i3.1662

Faizah, N. M., Surohman, Fabrianto, L., Hendra, & Prasetyo, R. (2020). Unbalanced Data Clustering with K-Means and Euclidean Distance Algorithm Approach Case Study Population and Refugee Data. *Journal of Physics: Conference Series*, *1477*(2). https://doi.org/10.1088/1742-6596/1477/2/022005

Garang, B. D. (2022). *Penerapan Data Mining Untuk Prediksi Penjualan Smartphone Paling Laris Menggunakan Metode K-Nearest Neighbor (Studi Kasus : Pusat Ponsel & Laptop)*. 1–54.

Informatika, S., & Polinema, A. (2020). Evaluasi Kmeans Clustering pada Preprocessing Sistem Temu Kembali Informasi. *Siap)*, 2020.

Jollyta, D., Efendi, S., Zarlis, M., & Mawengkang, H. (2019). Optimasi Cluster Pada Data Stunting: Teknik Evaluasi Cluster Sum of Square Error dan Davies Bouldin Index. *Prosiding Seminar Nasional Riset Information Science (SENARIS)*, *1*(September), 918. https://doi.org/10.30645/senaris.v1i0.100

Kurniadi, D., Agustin, Y. H., Akbar, H. I. N., & Farida, I. (2023). Penerapan Algoritma k-Means Clustering untuk Pengelompokan Pembangunan Jalan pada Dinas Pekerjaan Umum dan Penataan Ruang. *Aiti*, *20*(1), 64–77. https://doi.org/10.24246/aiti.v20i1.64-77

Listiani, L., Agustin, Y. H., & Ramdhani, M. Z. (2019). Implementasi algoritma k-means cluster untuk rekomendasi pekerjaan berdasarkan pengelompokkan data penduduk. *Seminar Nasional Sistem Informasi Dan Teknik Informatika*, 761–769.

Mathur, P. (2019). *Machine Learning Applications Using Python*. Apress.

Muliono, R., & Sembiring, Z. (2019). Data Mining Clustering Menggunakan Algoritma K-Means Untuk Klasterisasi Tingkat Tridarma Pengajaran Dosen. *CESS (Journal of Computer Engineering, System and Science)*, *4*(2), 2502–2714.

Nurjanah, M., & Arifin, T. (2021). Penerapan Algoritma K-Means Untuk Analisis Data Ulasan Di Situs Tripadvisor. *Jurnal Responsif : Riset Sains Dan Informatika*, *3*(1), 75–82. https://doi.org/10.51977/jti.v3i1.395

Orisa, M. (2022). Optimasi Cluster pada Algoritma K-Means. *Prosiding SENIATI*, 430–437. https://doi.org/10.36040/seniati.v6i2.5034

Ozdemir, S. (2017). *Principles of Data Science*. Packt Publishing Ltd. https://doi.org/10.1145/3097983.3105808

Paembonan, S., & Abduh, H. (2021). Penerapan Metode Silhouette Coefficient untuk Evaluasi Clustering Obat. *PENA TEKNIK: Jurnal Ilmiah Ilmu-Ilmu Teknik*, *6*(2), 48. https://doi.org/10.51557/pt_jiit.v6i2.659

Priyatman, H., Sajid, F., & Haldivany, D. (2019). Klasterisasi Menggunakan Algoritma K-Means Clustering untuk Memprediksi Waktu Kelulusan Mahasiswa. *Jurnal Edukasi Dan Penelitian Informatika (JEPIN)*, *5*(1), 62. https://doi.org/10.26418/jp.v5i1.29611

Purba, Y., Prayudha, J., & Azanuddin, A. (2022). Penerapan Metode K-Means Clustering Pada Data Mining Untuk Menentukan Genre Musik Lagu Di Radio Joy 101 Fm. *Jurnal Cyber Tech*, 1–8. https://ojs.trigunadharma.ac.id/index.php/jct/article/view/1646%0Ahttps://ojs.trigunadharma.ac.id/index.php/jct/article/download/1646/1002

Quinthara, D. R., Fauzan, A. C., & Huda, M. M. (2023). Penerapan Algoritma K-Modes Menggunakan Validasi Davies Bouldin Index Untuk Klasterisasi Karakter Pada Game Wild Rift. *Journal of System and Computer Engineering (JSCE)*, *4*(2), 123–135. https://doi.org/10.61628/jsce.v4i2.802

Sholeh, M., & Aeni, K. (2023). Perbandingan Evaluasi Metode Davies Bouldin, Elbow dan Silhouette pada Model Clustering dengan Menggunakan Algoritma K-Means. *STRING (Satuan Tulisan Riset Dan Inovasi Teknologi)*, *8*(1), 56. https://doi.org/10.30998/string.v8i1.16388

Tambunan, M. P. (2021). Penerapan Data Mining Dalam Analisa Data Pemakaian Obat Dengan Menerapkan Algoritma K-Means. *Jurnal Informasi Dan Teknlogi Ilmiah (INTI)*, *8*(3), 109–113.

Vania, P., & Sari, B. N. (2023). Perbandingan Metode Elbow dan Silhouette untuk Penentuan Jumlah Klaster yang Optimal pada Clustering Produksi Padi menggunakan Algoritma K-Means. *Jurnal Ilmiah Wahana Pendidikan*, *9*(2), 547–558.