UNIVERSITAT DE BARCELONA

Work 2: Dimensionality Reduction and Visualization

Introduction to Machine Learning

Hasnain Shafqat

João Valério

Eirik Grytøyr

Date: 13/11/2022

Index

1. INTRODUCTION

The main goal of the present work is to study the topics of dimensionality reduction and data visualization, using three datasets from the UCI repository.

The collection of datasets utilised remains unchanged from the previous work. These are SatImage, Hepatitis and cmc, respectively, with the following characteristics: only numerical with significant size, numerical and categorical with small size, and, numerical and predominantly categorical with relevant size.

In the dimensionality reduction matter, the 3 distinct types of algorithms analysed are Principal Component Analysis (PCA), Incremental Principal Component Analysis (IPCA) and Feature Agglomeration (FA). The mentioned are employed considering sklearn's library and, in addition, PCA is as well self-implemented. Through that, the aim is to comprehend the influence of dimensionality reduction techniques directly in the data and in the posterior clustering quality performed by K-Means (self-implemetation) and Agglomerative Clustering (sklearn). In order to quantify the outcomes of these methodologies, it is computed the correspondent confusion matrices, F1-Scores and 3D visualisations.

The visualization process is conducted by Principal Component Analysis (PCA), T-distributed Stochastic Neighbor Embedding (t-SNE) and the absence of dimensionality reduction, in the application of K-Means and Agglomerative Clustering. Thus, the scatter plots describe illustratively the clusterings performed, with the objective of perceiving the effect of distinct approaches in dimensionality reduction. All the plots can be found in Appendix A.

Finally, as an overall goal, it is pretended to improve the results accomplished in the previous project, via the correct selection of components.

2. ANALYSIS

In the present chapter, it is developed the analysis of the present study in five distinct units. Point a. evaluates the effect of the self-implementation of PCA in the datasets of the current study (SatImage, Hepatitis and cmc). Then, in unit b. it is developed a comprehension of the main differences between Principal Component Analysis (PCA) and Incremental Principal Component Analysis (IPCA) so that in c, it is possible to understand the impact of PCA in clustering algorithms, such as K-Means (self-implementation) and Agglomerative Clustering (sklearn library). Point d. is a repetition of b. and c., in which the algorithm applied to execute dimensionality reduction is Feature Agglomeration (sklearn library). In e. the clusters from K-Means and Agglomerative Clustering are visualized recurring to PCA, T-distributed Stochastic Neighbor Embedding (t-SNE) and dimensionality reduction's absence (work 1). As a final note, in this assignment, the preprocessing technique One-Hot Encoding was replaced by Label Encoding, since with the first the feature space increases considerably and, consequentially, some eigenvectors would be constituted by complex numbers

a. PCA algorithm with the datasets

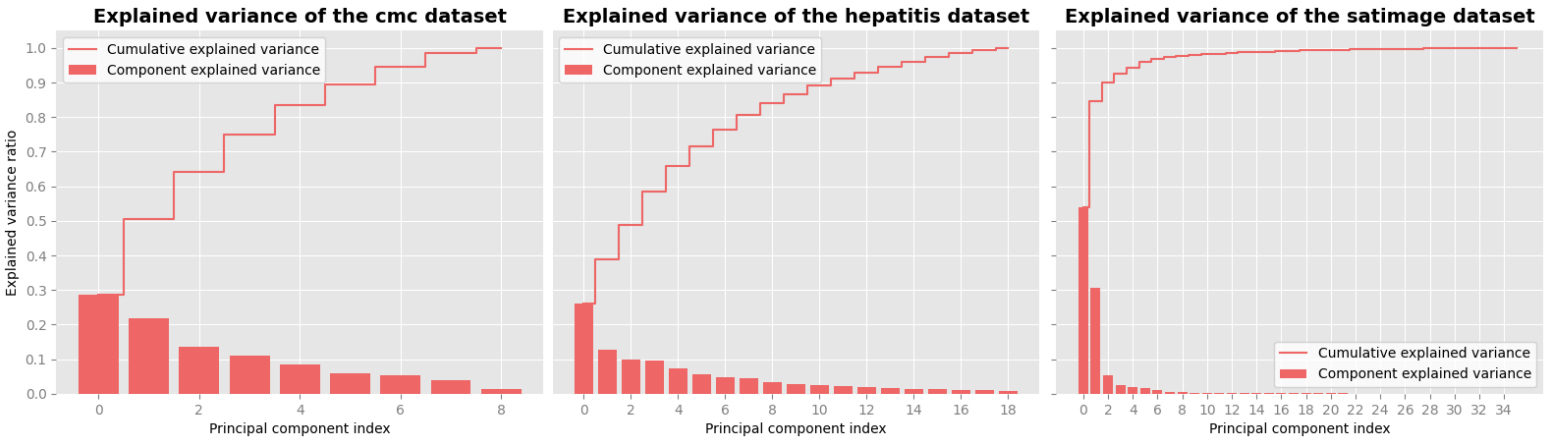
In this first section, it is analyzed the implementation of the self-developed PCA algorithm in the three datasets mentioned. Firstly, it is important to refer that the main setup of PCA is the following:

1. Compute the d-dimensional mean vector;
2. Compute the covariance matrix of the entire data;
3. Calculate the eigenvectors and the correspondent eigenvalues;
4. Sort the eigenvectors by decreasing eigenvalues;

5. Choose k eigenvectors with the largest eigenvalues to form a new d x k dimensional matrix;
6. Finally, derive the new data set.

Then, in accordance with the previous explanation, the analyses occur recurring to helpful plots to understand in more detail the PCA algorithms results. To begin with, the explained variance by the components of the datasets (figure 1) is the primary study to be performed.

Figure 1 – Explained variances of the 3 datasets.

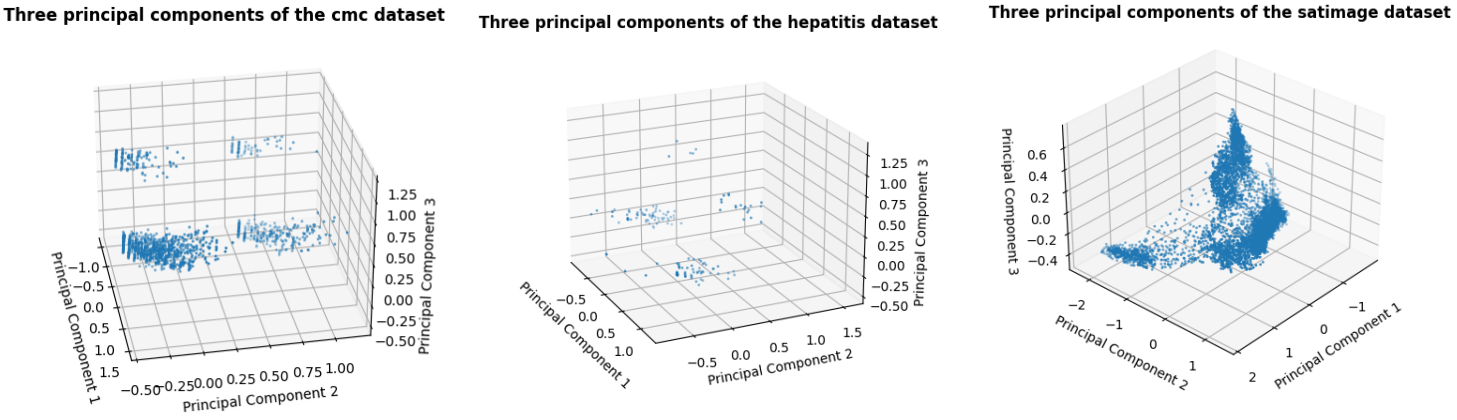


Firstly, it is observable that while Hepatitis and cmc datasets graphs are similar, SatImage presents a distinct disposition. Notably, the difference between consecutive components is less notorious in the first 2 datasets, when compared to the latter. In fact, in SatIamge with only 4 components, it is possible to achieve 90% of the variance, indicating that the main underlying information of the data is contained in a diminutive set of features. On the contrary, in the cmc and Hepatitis datasets, 6 and 11 components are necessary, respectively, because the information is more distributed among the features.

Quantitatively, selecting 90 % of the total variance permits a considerable reduction in the number of features in almost all three datasets. In satimage the reduction goes from 35 features to just 3, in which, according to the graph, it can be foreseen a good clustering of the data. Moreover, in the Hepatitis dataset, the discrepancy is 7 (from 19 to just 12 features) and, finally, cmc registers the lower drop, which is from 9 features to 7. Overall, this reduction in dimensionality can help discard irrelevant features that add computational time, through the additional dimensions. However, as will be seen in the subsequent chapters, the different explanation variance ratio dispositions lead to distinct clustering results.

In order to complement the previous illustration, it is plotted the 3 most important components of each dataset in figure 2, by reducing the dataset with 90 % of explained variance.

Figure 2 – The 3 principal components of each dataset.



By analysing the previous figure, it is understandable that not all the real number of clusters are evidently identifiable. For example, in cmc plot, it is apparently observable a set of 4 groups of points, while in fact, the real labelled number is 3. The same pattern is visualized in the Hepatitis plot. This indicates that more (or other) features would be needed to extract the correct information relative to the clustering, sustaining the information provided by figure 1.

In SatImage, the illustration obtained corresponds to the complete description, since the data was reduced to 3 components. However, the density of the data points does not allow for suitable visual conclusions. This constraint is in accordance with the analysis of the previous work, in which the internal clustering metrics identified 3 clusters, instead of 6. Even now, it is visually imperceptible the real outcome, which would not be achievable without the ground truth help.

Finally, in general, it is comprehended the importance of the dataset's visual representation with the most critical components, since the perception is more aligned with the factual information of the data.

b. Dimensionality Reduction with PCA and IPCA

In the present chapter, it is executed 3 algorithms with 2 approaches, Principal Component Analysis (PCA) and Incremental Principal Component Analysis (IPCA), in order to perform dimensionality reduction. Both are implemented according to the sklearn.decomposition's library and, additionally, PCA is also self-implemented with respect to the requirements of the assignment.

In order, to apply PCA to each dataset of the present study (SatImage, Hepatitis and cmc), it is crucial to define priorly the number of components. In this selection, it is considered figure 1 from chapter a., which enables an understanding of the most substantial components in the dataset. Furthermore, it is also desirable that the values cover a wide scope of dimensionality. According to that, the following table represents the information considered regarding the components.

Table 1 – Components of each dataset.

	SatImage			cmc				Hepatitis			
Number of Components	10	5	3	7	5	4	2	15	10	5	3

Theoretically, IPCA is used in replacement of the PCA, when the dataset is substantially large for the memory's capacity, through the adjustment of the batch size. Therefore, IPCA constitutes an approximation of PCA, in which the batch size defines the trade-off between memory usage and accuracy. Mainly, in more undersized datasets, the approximation effect is more noticeable. In the current study, the greater dataset is SatImage with 6435 samples and 36 features, which, normally, would not require the application of IPCA, as well as the remaining data sets.

Firstly, it is demonstrated, through table 2 of SatImage, that the memory used in the PCA is larger than in IPCA applications.

Table 2 – Memory used by PCA and IPCA in SatImage dataset.

	10 Components		**5 Components**		**3 Components**	
Algorithm	**PCA**	**IPCA**	**PCA**	**IPCA**	**PCA**	**IPCA**
Memory Used [GB]	1.4374	1.4273	1.4043	1.3889	1.4040	1.3492
Δ [GB]	0.0101		0.0154		0.0548	

However, as stated previously, even though SatImage is the largest dataset, it does not justify the usage of IPCA, as can be seen by the reduced differences between both algorithms. Precisely, both operations only use roughly 9% of the RAM. Furthermore, it is observed that the memory used decreases and the discrepancy between algorithms increases (different slopes) along with the decrement in the number of components. Finally, for the spectrum of data sizes considered in this study, it can be noted that both algorithms have, roughly, similar time complexities.

By executing each algorithm, it is comprehended that the disparities are significantly reduced. As an illustration of the eigenvalues obtained for the different algorithms, table 2 represents, with 3 components, Hepatitis, cmc and SatImage datasets, the smallest (155 samples and 19 features), medium (1473 samples and 9 features) and biggest (6435 samples and 36 features), respectively.

Table 3 – Eigenvalues of Hepatitis and SatImage with 3 components.

Data	**Algorithm**	**Eigenvalues**		
Hepatitis	**PCA**	0.32054774	0.15776284	0.12125135
	PCA - sklearn	0.32054774	0.15776284	0.12125135
	IPCA - sklearn	0.30935356	0.15391945	0.10123123
	|PCA - IPCA|	0.01119418	0.00384339	0.02002012
cmc	**PCA**	0.24172670	0.18503883	0.11505480
	PCA - sklearn	0.24172670	0.18503883	0.11505480
	IPCA - sklearn	0.24137162	0.18478887	0.10343852
	|PCA - IPCA|	0.00035508	0.00024996	0.01161628
SatImage	**PCA**	0.70574132	0.39983817	0.06901990
	PCA - sklearn	0.70574132	0.39983817	0.06901990

	IPCA - sklearn	0.70543424	0.39903088	0.06473465
	|PCA - IPCA|	0.00030708	0.00080729	0.00428525

According to the values exposed in the previous table, it is possible to conclude that the self-implementation of PCA was successfully executed since the eigenvalues are identical to the sklearn ones. Moreover, this pattern is transversal to the remaining number of components of table 1. An important note is that, even though both PCA algorithms achieve the same eigenvalues, occasionally, the spatial orientation of the eigenvectors differs.

On the other hand, it is detected a slight contrast between the PCA and IPCA algorithms, denoting the latter's approximation effect. Particularly, it is demonstrated the theoretical relation between IPCA and the size of the dataset, since the discrepancy |PCA-IPCA| is larger in Hepatitis, the smaller dataset, than in SatImage or cmc. Moreover, usually, the mentioned difference is likewise more significant in cmc than in SatImage, however, the dissimilarities are not as substantial as the previous ones. Thus, IPCA works better in larger datasets, proving to be a suitable alternative, in the presence of memory constraints.

Furthermore, as the eigenvalues obtained by IPCA's algorithm are also smaller than PCA, it is expected and verified that the explained variance ratio is also more diminutive in IPCA. In other words, the approximations effectuated by the algorithm are done by default (underestimation). As a consequence of this type of approximation, the estimated noise covariance is also lower in IPCA due to the less effect of the method in the dimensionality reduction process.

c. PCA's Dimensionality Reduction and Clustering

In the prior chapter, it was inferred that for the current application, PCA is more suitable as a dimensionality reduction tool. So, in order to give continuity to PCA's study, it is pretended to comprehend the results achieved by K-Mean (self-implementation) and Agglomerative Clustering (sklearn) in the original and dimensionally reduced datasets.

In the evaluation process, to obtain the confusion matrices along with the F1-Scores, it is utilised the real number of clusters per dataset (SatImage 6, cmc 3 and Hepatitis 2) and, when suitable, an explained variation ratio of 0.9, corresponding to 4, 7 and 12 components in SatImage, cmc and Hepatitis, respectively.

Figures 3 corresponds to K-Means algorithmic results when dimensionality reduction is not applied, as in the previous work, while in figure 4 PCA is previously applied to the data.

Figure 3 – K-means clustering without dimensionality reduction.

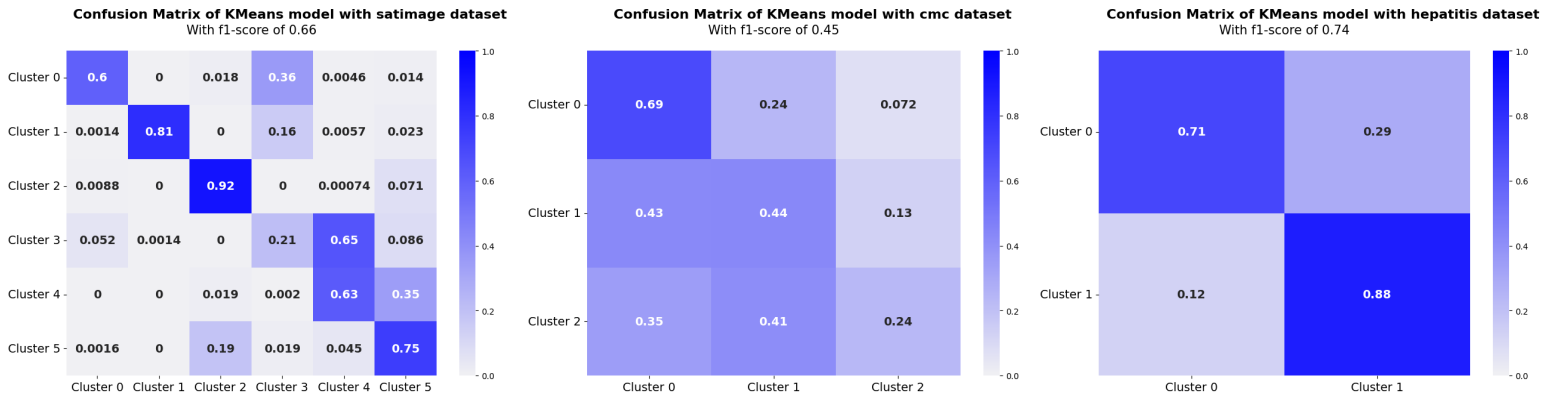
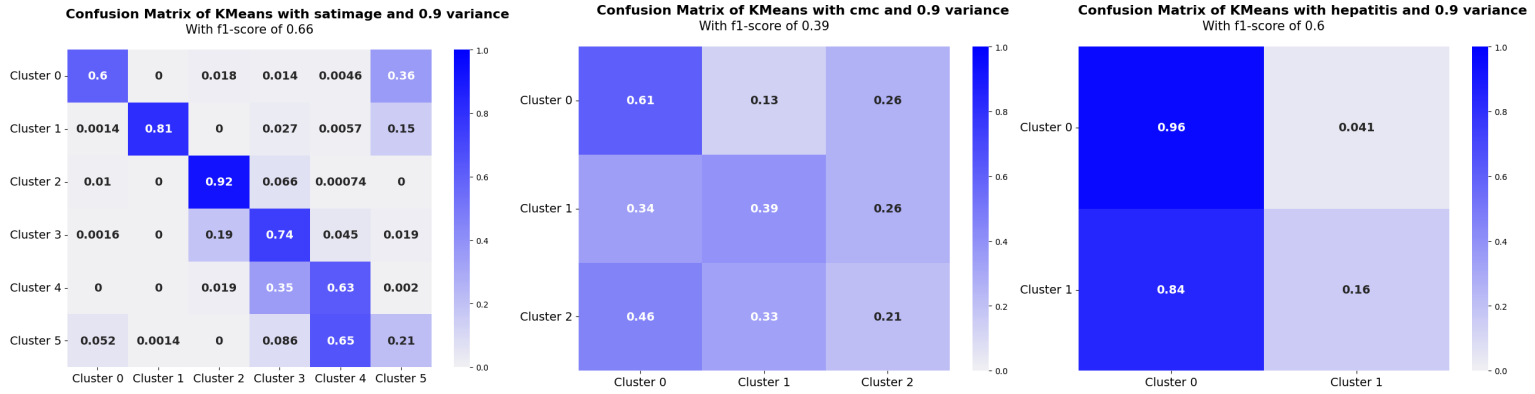


Figure 4 – K-means clustering with PCA dimensionality reduction.



Accordingly to the results obtained, in general, the F1-Scores decrease when PCA is applied priorly to the clustering, with the exception of SatImage. In this dataset, the score of 0.66 remained identical, even though the clustering dispositions are distinct. This result is mainly due to the size (6435 samples and 36 features) and balance (Majority class - 23.82% and Minority class - 9.73%) of the dataset, in which the relevancy or impact of the information lost is lower than in the remaining ones. Particularly, in the true positive identifications (main diagonal), only clusters 3 and 5 were significantly affected in the same scale, in which the first improved, while the latter deteriorated. In consonance with the prior analysis, it is conceivable to indicate that SatImage is a suitable dataset to apply dimensionality reduction, without interfering greatly with the quality of the final results.

On the opposite, cmc and Hepatitis are smaller (1473 samples and 9 features - 155 samples and 19 features, respectively) and more unbalanced (majority class of 42.70% and minority class of 22.61% - majority class of 79.35% and minority class of 20.65%, respectively) datasets. Thus, the effect of PCA on the information is more notable. However, while in cmc the influence is approximately distributed among the classes, in Hepatitis the majority of the data points were considered as members of cluster 0. So, perhaps the main information under the pattern for class 1 members was lost, resulting in the deterioration of the outcomes.

Figures 5 and 6 represent a similar analysis for the Agglomerative Clustering algorithm.

Figure 5 – Agglomerative Clustering without dimensionality reduction.

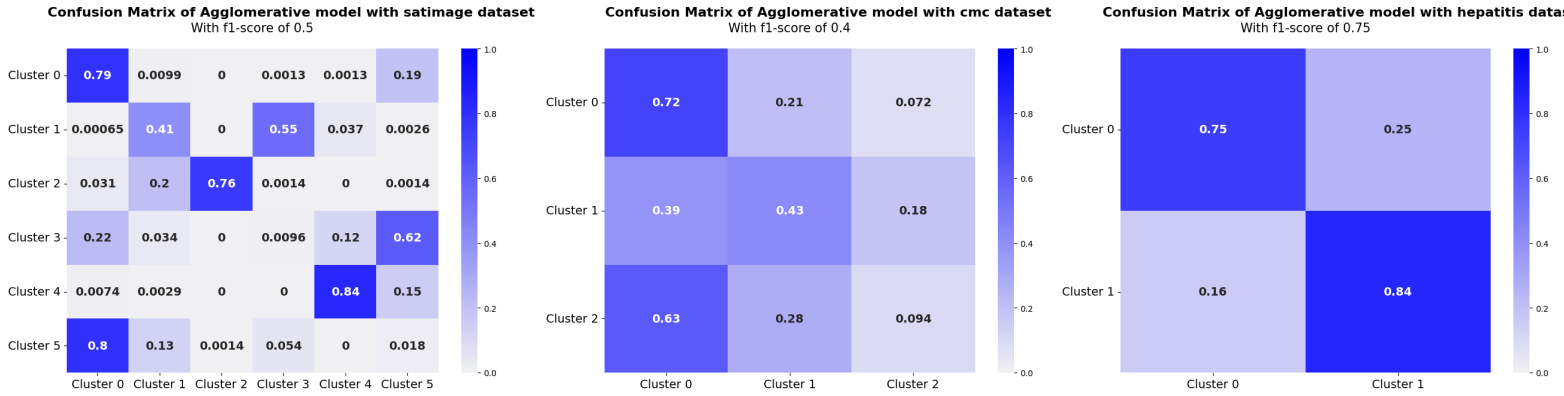
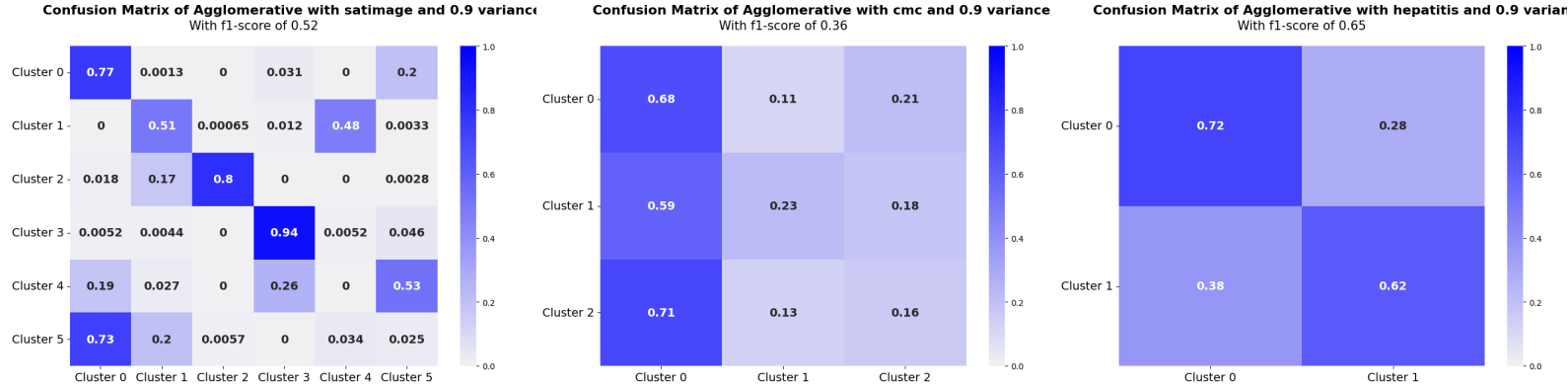


Figure 6 – Agglomerative Clustering with PCA dimensionality reduction.



In Agglomerative Clustering, the overall patterns for cmc and Hepatitis datasets are equivalent to K-Means, but with fewer results deterioration. However, in particular, it is acknowledged some differences. Firstly, in cmc, not all the clusters are impacted roughly on the same scale as in K-Means. Instead, cluster 1 presents a considerable downward tendency in the score, while cluster 2 identification improves. Then, in Hepatitis both clusters achieve worse results, in which once again cluster 1 is the most influenced. As seen previously, the misclassifications of cluster 1 grow, but more slightly this time. Overall, the results for both datasets are justified by the size and unbalance of the data, as mentioned before.

This time, SatImage registered an improvement from 0.50 to 0.52 in the F1-Score, in which all the clusters were seemingly impacted by the PCA. Particularly, clusters 3, as seen priorly, and 4 were the most affected. While the first increased dramatically the results, the latter went from 0.84 to 0.00. In other words, Agglomerative Clustering lost completely the capacity of identifying members of cluster 4. These local extreme variations seen in figures 4 and 6 might be due to the fact that with PCA only 3 features are being considered. Nonetheless, even though these are the most relevant generally, might occur that a particular cluster depends heavily on a specific feature and, consequentially, on the PCA transformation process. Finally, overall, it is notable an elevated influence of the three main features, illustrated graphically in figure 1.

According to the prior analysis, PCA was demonstrated to be a useful tool to perform dimensionality reduction, in which there is a trade-off between information and dimension. However, its applicability and utility depend immensely on the dataset and the final aims, since it works competently in large and balanced datasets, such as SatImage, but not in small and unbalanced ones, like cmc or Hepatitis. Finally, the reduction of dimensionality permits achieving the final clusters with less time complexity.

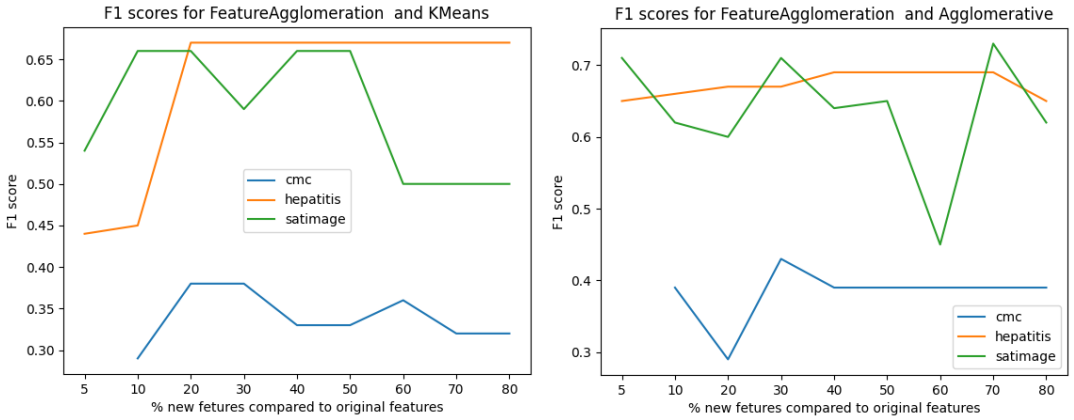
d. Feature Agglomeration's Dimensionality Reduction and Clustering

In the current chapter the Feature Agglomeration algorithm from the sklearn library is examined and compared with the PCA results in section c. Particularly, different k-values of the Feature Agglomeration will be tested with respect to the F1-Scores obtained from the confusion matrix. Then, the most satisfactory k-values will further be used to compare the F1-Scores and discuss the visual representation of the clusters, for the results of both dimensionality reduction algorithms.

Feature agglomeration uses the hierarchical clustering method to group the most similar features together where k is the number of groups [1]. Since the algorithm doesn't provide the explained

variance to decide the number for k, the F1-Scores for each dataset with different k-values are compared as shown in Figure 7.

Figure 7 – F1 scores as functions of k represented by the percentage of features.



The x-axis represents the percentage of features left from the original data, which is used as the k-value. For instance, in the Hepatitis data set, 10 % of the features indicate a k-value of 2. Additionally, as in chapter c, the clustering is accomplished regarding the real number of clusters.

Firstly, due to the continuous oscillations, the illustration does not indicate a clear trend for all the datasets on whether a larger or smaller dimensionality produces a more suitable clustering. With agglomerative clustering, the lowest and highest values obtained for cmc are extremely close, respectively, 20% and 30% of the features. In fact, this pattern is transversal to the SatImage dataset, in which the lowest score corresponds to a k representing 60 % of the features and the soundest score is at 70 %. This demonstrates that the merge of features with the largest similarity is not necessarily the best way to maintain the representation of the differences in the data, since a merge of two features can lead to large differences.

Table 4 exhibits the best results obtained in the previous analysis along with the values obtained in work 1. In the parenthesis, it is displayed the parameter k (number of clusters) used in feature agglomeration and its rate compared to the number of original features.

Table 4 – F1-score.

		KMeans		Agglomerative	
		Without dimensional reduction	**With Feature Agglomeration**	**Without dimensional reduction**	**With Feature Agglomeration**
Data	**cmc**	0.45	0.38 (2/20%)	0.4	0.43 (3/30%)
	Hepatitis	0.74	0.67 (4/20%)	0.75	0.69 (8/40%)
	SatImage	0.66	0.66 (4/10%)	0.5	0.73 (25/70%)

According to the outcomes present in the table, the Agglomerative Feature reduction did not improve the clustering results for the KMeans algorithm. In fact, K-Means indicates the best results for very few features in all the datasets, which had only been suitable for SatImage in chapter c. Particularly, by

comparing the F1-Scores obtained with the ones accomplished by PCA in chapter c., the F1-Scores' evolution is similar.

On the other hand, it is noted an improvement in Agglomerative Clustering, in SatImage and cmc datasets, which, for instance, are the larger ones. Taking into account point c., this time the results are not exactly equal, since inserting PCA decreased the F1-Score of cmc. Furthermore, the improvement registered in the SatImage dataset is more sound (23%), indicating that the original data is constituted by a fair amount of non-relevant information for the clustering. Finally, based on the average score for the clustering algorithms, Feature Agglomeration is able to perform a slight improvement of 6 % compared to PCA.

According to the plots of PCA and Agglomerative Clustering in Appendix A, the discrepancies are unnoticeable. The same dense groups are clustered, however, in the conditions where the groups are split between the labels, the algorithms distinguish a bit in the boundary between the clusters. Furthermore, as in PCA, the impact of Agglomerative Clustering is different depending on the clusters.

Table 5 shows the time consumption of the Feature Agglomeration with different numbers for k.

Table 5 – 1000 Time usage average from the feature agglomeration for the SatImage data set.

	4 features	**8 features**	**12 features**
Time consumption [ms]	36	38	41

Table 5 reveals that, as stated in c, the time consumption increases along with the number of features selected, even though it means fewer features to combine. Quantitatively, the algorithm has a good time performance, which makes the measure of time less accurate but indicates that it is efficient to reduce the number of features to improve the time consumption of a learning algorithm.

To sum up, by considering the F1-Scores and the visual examination of the clusterings, the performance of Feature Agglomeration is satisfactory and not extremely different from PCA. However, for medium-sized datasets, the present one achieves better results.

e. Visualisation with PCA and t-SNE

In order to visualise the data in a low dimensional space, the two different approaches tested are PCA and T-distributed Stochastic Neighbor Embedding (t-SNE). All the plots can be found in appendix A, with the respective K-parameters. The additional parameters in t-SNE are "perplexity" and "early_exaggeration", which were tested with a +- 50% change. Even though it verified an impact on the representation, there was not a clear improvement. Thus, the default parameters were selected.

According to Appendix A, the PCA and t-SNE have distinct forms of representing the data. The t-SNE spreads the clusters in a larger area of the space and makes it easier to notice the different patterns in the data. It is also mentioned that with the 3D representation in the report it can be difficult to see the depth of the figure. With the code provided, it is possible to drag and change the orientation of the plot. When this is done, it is easier to observe the clusters in the t-SNE plots than in the PCA plots. An exception is for the hepatitis data set. In this situation, the clusters are not distinguishable in the t-SNE plot as opposed to the PCA. This might be because the t-SNE is especially suitable for larger data sets.

In appendix A, the rows represent different dimensional reduction methods before applying the algorithm. It is observed that when the same method is applied for dimensional reduction and the visualisation achieves better results. This suggests that while choosing the method for visualisation, it is

important to take into account which dimensional reduction method is used priorly to the clustering algorithm.

A downside of using t-SNE is the computational expensiveness of the algorithm. For instance, the PCA algorithm operates in a few milliseconds for the Hepatitis and SatImage datasets, while t-SNE, on the other hand, utilised, approximately, 2 and 160 seconds for the same data sets. This can be an important consideration when working with a larger data set. This way T-SNE might not be a useful way to reduce the total time consumption in a clustering process, by decreasing the dimensionality.

3. CONCLUSION

All the goals initially proposed were achieved with success and will be explained during the conclusion.

Firstly, it was comprehended that PCA allows the reduction of the data's dimensionality by computing the most important features/components in the dataset, in which the difference between the consecutive components depends on the data. In fact, it was apprehended that keeping, for example, as much as 90% of the total variance results in a considerable reduction in the number of features, by discarding the irrelevant ones that contribute to supplementary computational time.

By analysing the behaviour of the algorithms PCA and IPCA, it was inferable that the outputs achieved are extremely similar, in which the discrepancy increases with the decrement of the data's size. However, it was perceptible that IPCA is a suitable tool for significantly large datasets.

Furthermore, PCA provides selectively the most important underlying information in a data set in a successful manner, resulting in a quicker clustering process afterwards. However, the applicability of the model depends on the inherent characteristics of the dataset. Particularly, it is more appropriate for large and balanced ones. So, depending on the data, any evolution in the quality of the outcomes is possible. Moreover, it was noticeable that usually the majority of the clusters are impacted, but at distinct levels.

Feature agglomeration shows similar results in F1-score and visually conserves the patterns in the data as for PCA, but it is observed a small improvement when agglomerative clustering is used. Mainly, cmc outcomes improve, even though this is not a large dataset.

For visualisation of the data, PCA and t-SNE achieve remarkably different outcomes. In general, PCA is preferable for dimensionality reduction, especially for larger data sets, because the time consumption is much lower. For data representation, the t-SNE algorithm provides a better distinction between the clusters and is preferable in large datasets.

To conclude, dimensionality reduction and visualization tools are extremely appropriate tools, in order to apply the clustering methods. However, a proper understanding of the data is required previously to improve the results.

4. BIOGRAPHY

[1] scikit-learn developers, “sklearn.cluster.FeatureAgglomeration” [accessed 2022-11-12] https://scikit-learn.org/stable/modules/clustering.html#hierarchical-clustering

[2] scikit-learn developers, “sklearn.decomposition.PCA” [accessed 2022-11-10] https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.PCA.html

[3] scikit-learn developers, “sklearn.Incremental.PCA” [accessed 2022-11-10] https://scikit-learn.org/stable/auto_examples/decomposition/plot_incremental_pca.html

[4] Medium, “Understanding PCA and T-SNE intuitively” [accessed 2022-11-10] https://medium.com/analytics-vidhya/understanding-pca-and-t-sne-intuitively-f8f0e196aee4

[5] JMLR, “Visualizing Data using t-SNE” [accessed 2022-11-10] https://www.jmlr.org/papers/volume9/vandermaaten08a/vandermaaten08a.pdf

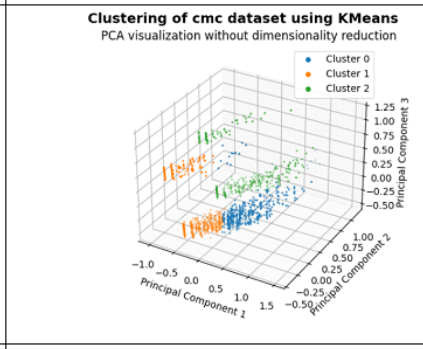
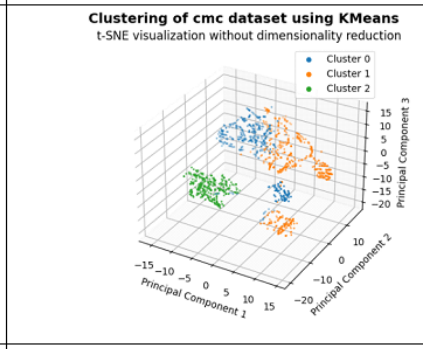
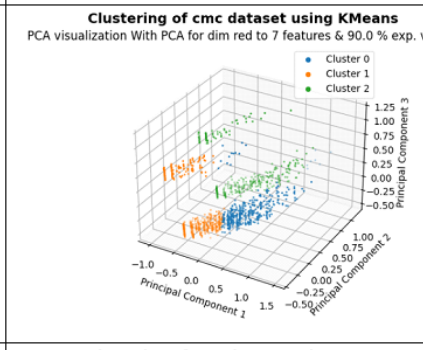
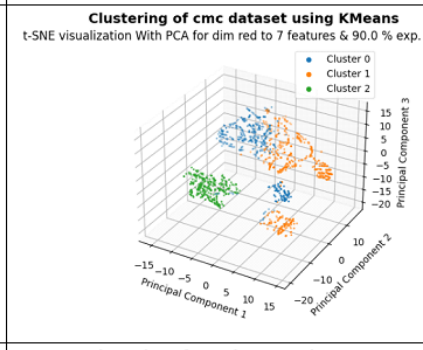
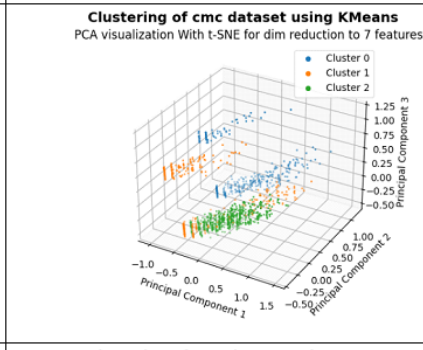
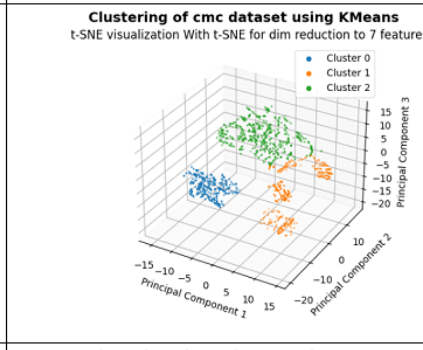
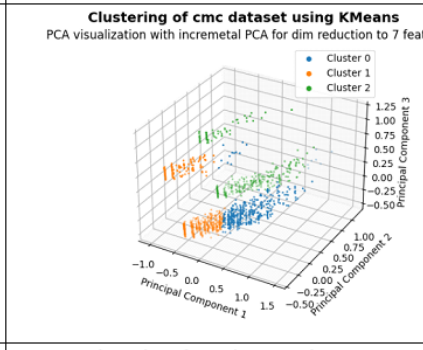
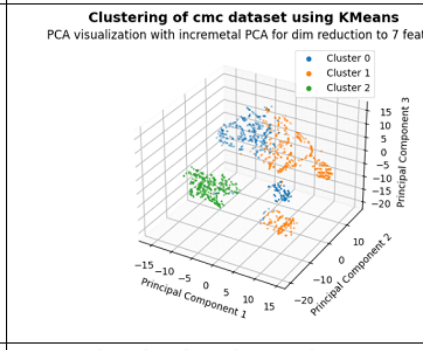
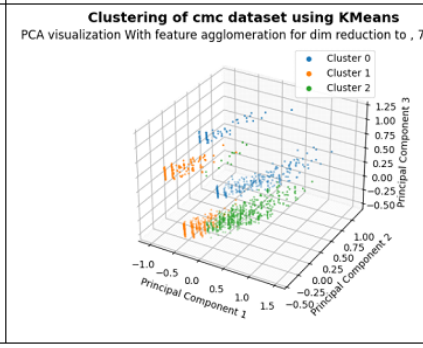
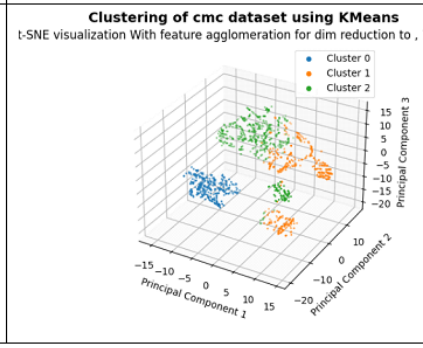
Appendix A

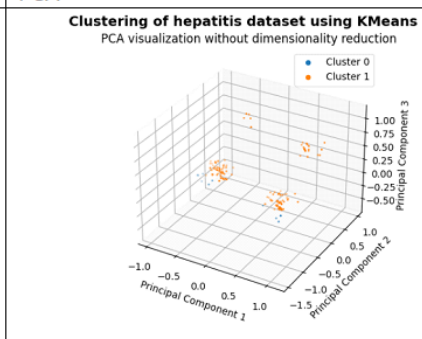
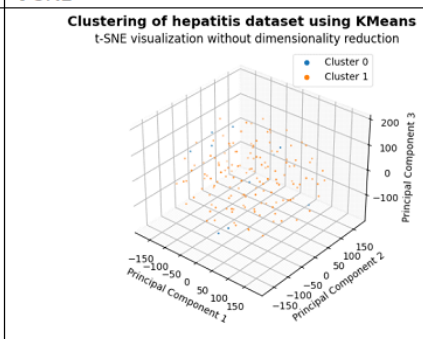
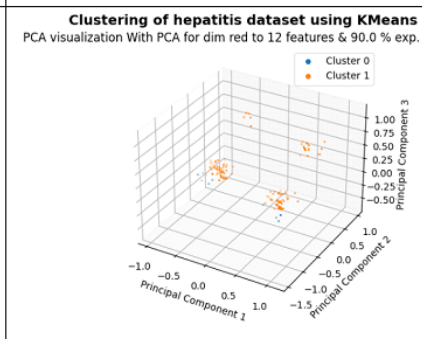
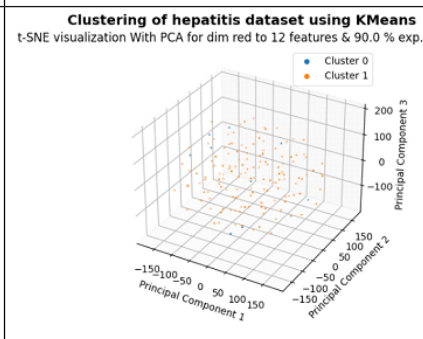
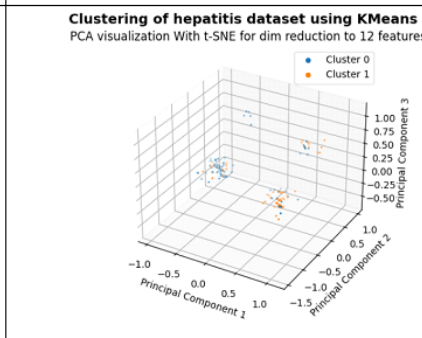
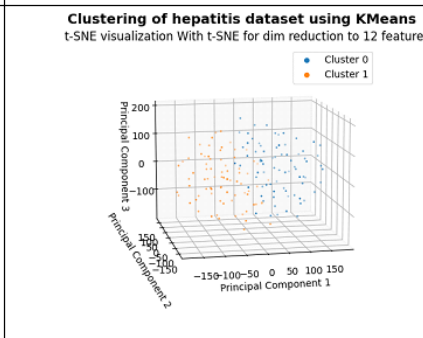
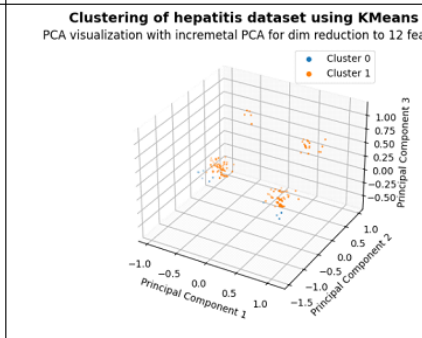
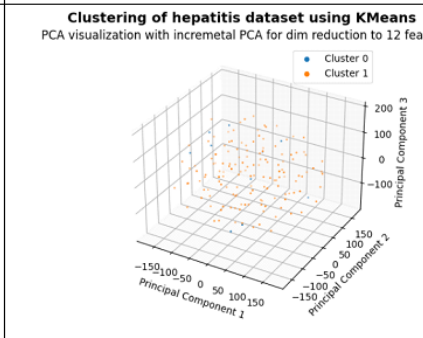
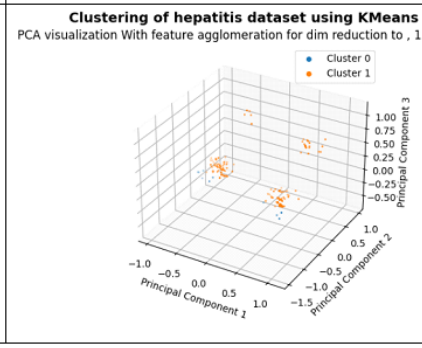
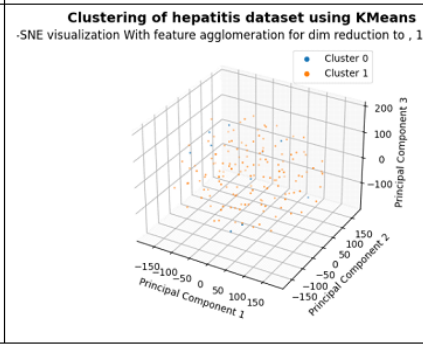
Plot of clusters

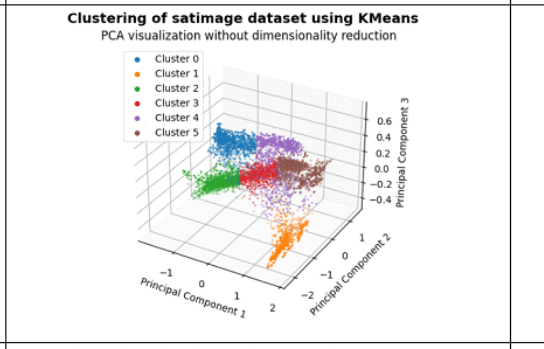
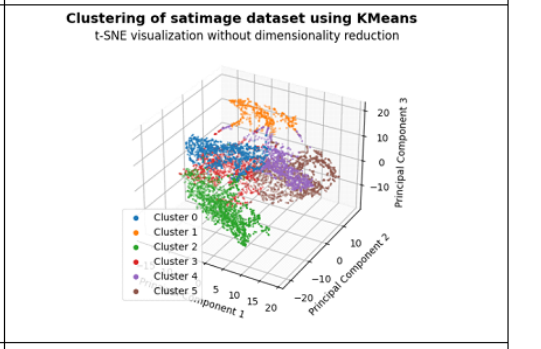
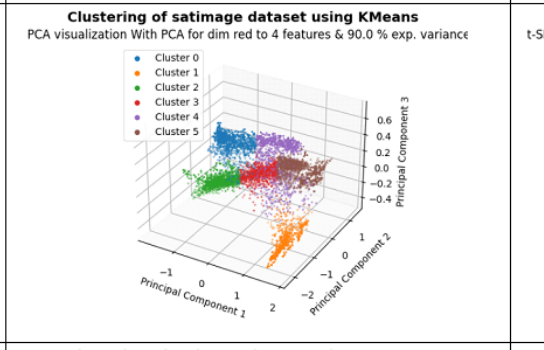
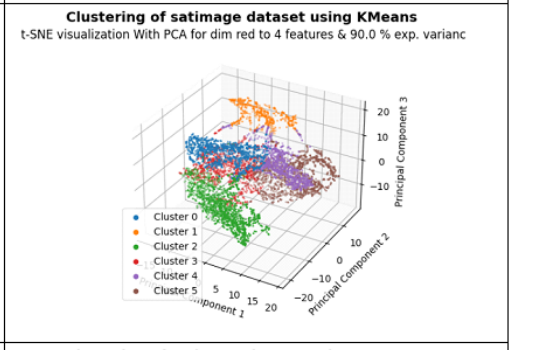
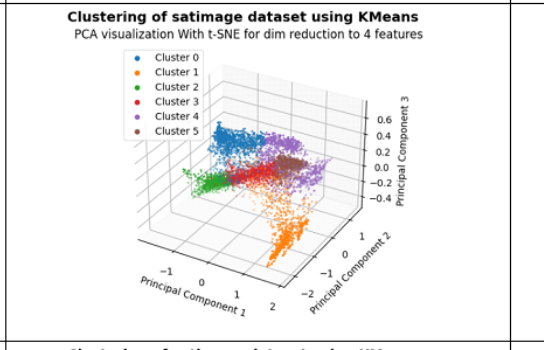
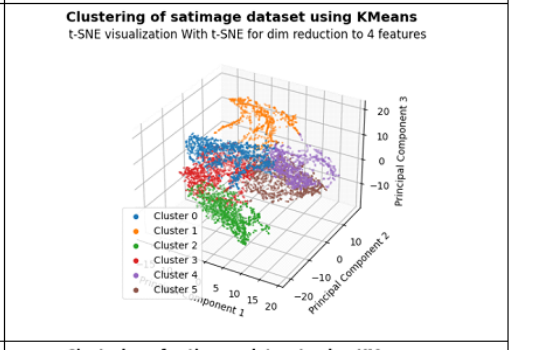
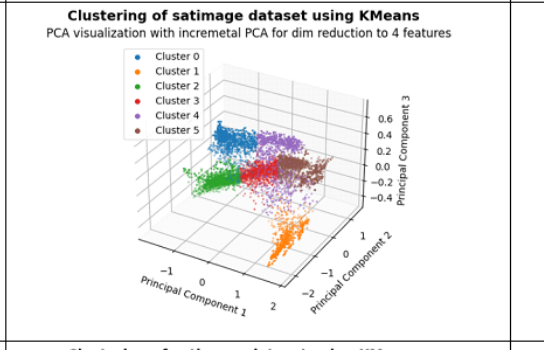
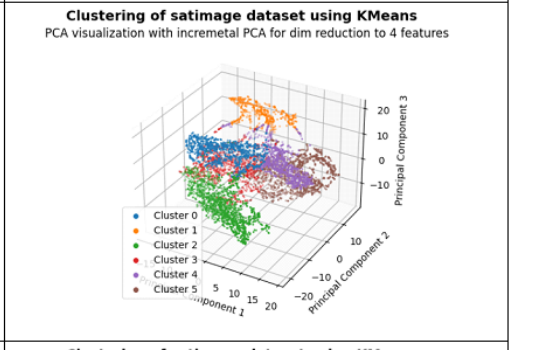
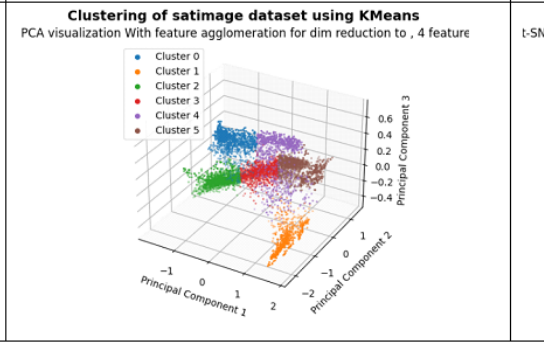
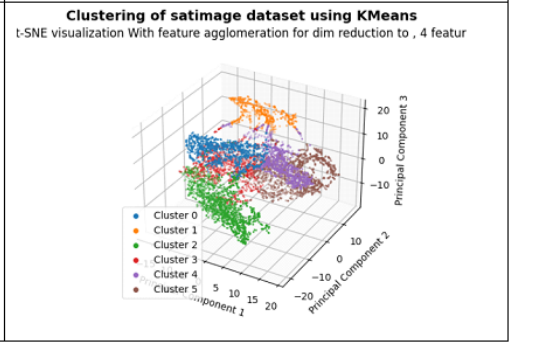
This Appendix contains 6 tables. One for each combination of clustering algorithm and data set.

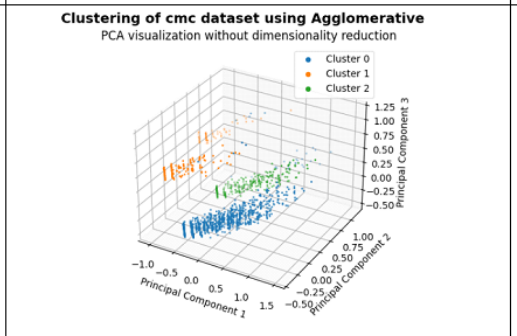
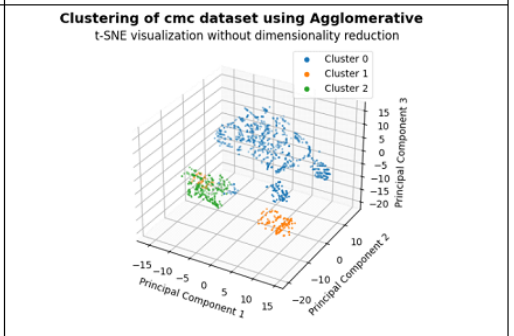
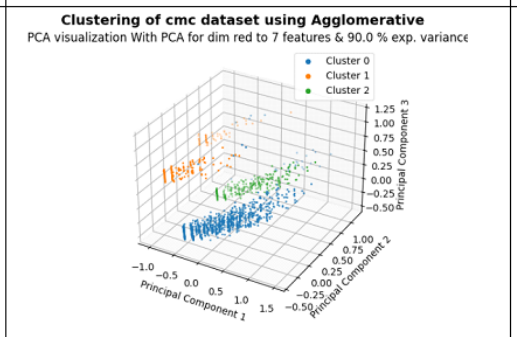
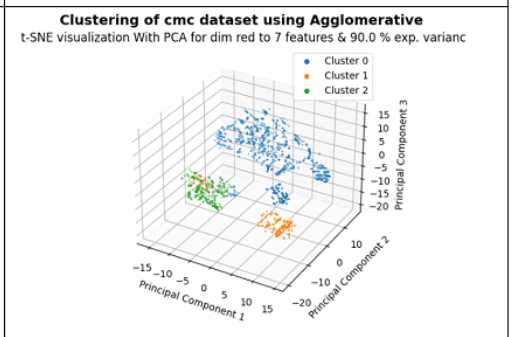
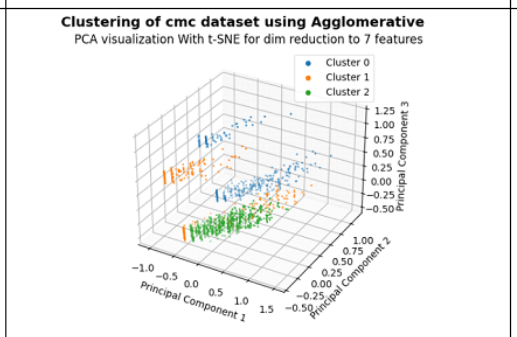
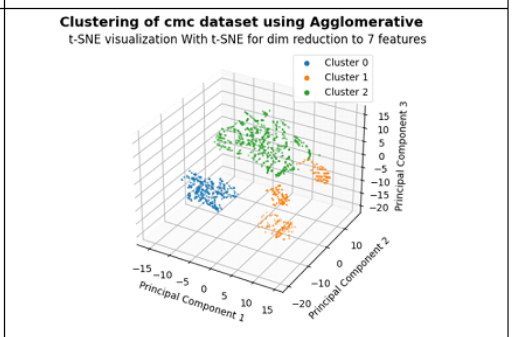
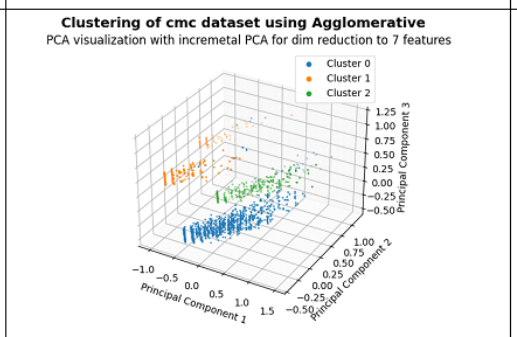
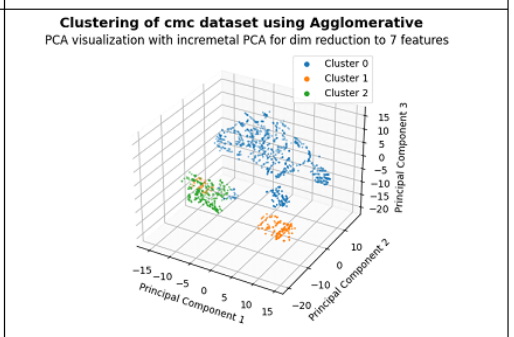
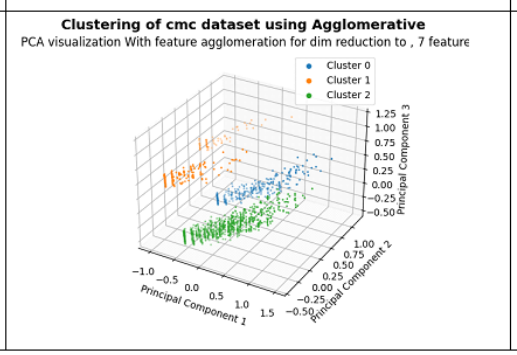
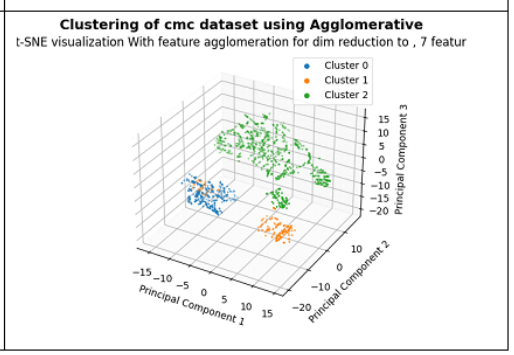
in the tables, the columns represent the algorithm used for data visualisation.

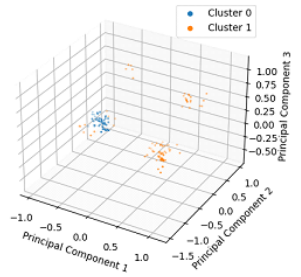
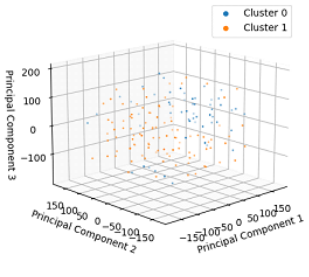
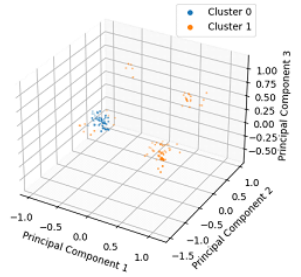
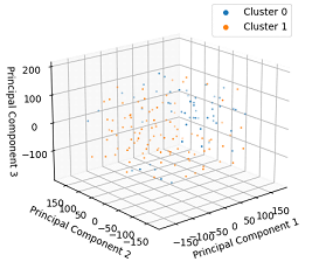
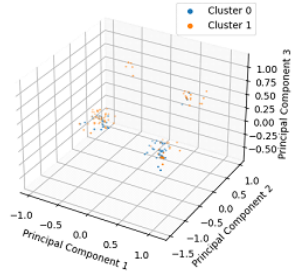
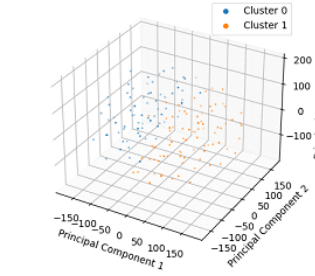
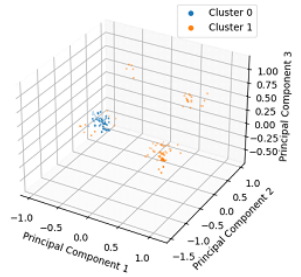
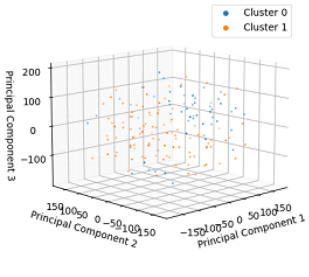
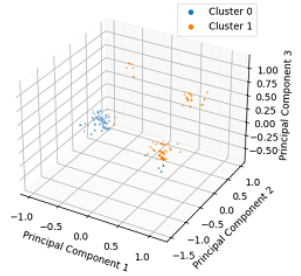
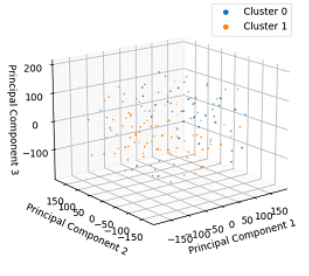
The rows represent the dimensional reduction algorithm used before clustering.

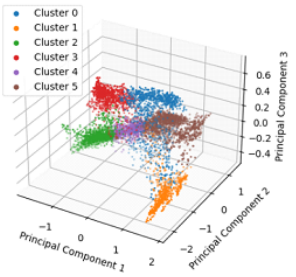
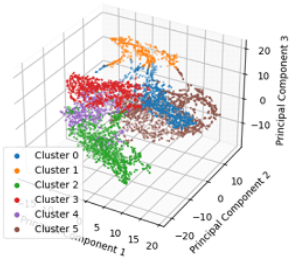
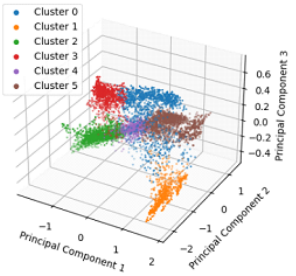
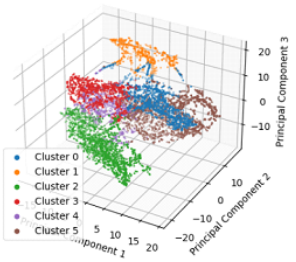
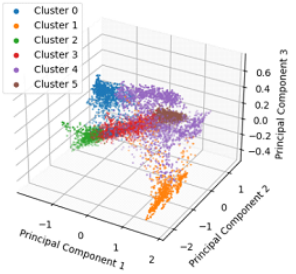
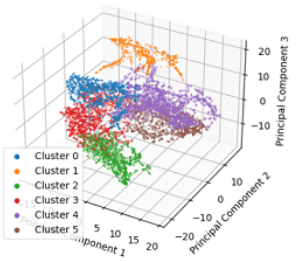
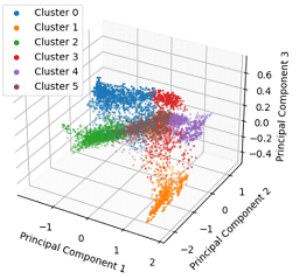
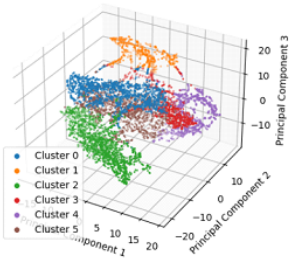
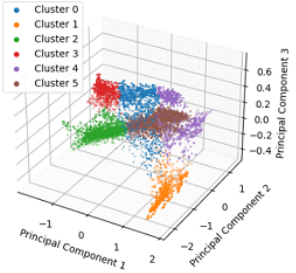
	cmc using KMeans	
	PCA	t-SNE
Without feature reduction		
pca		
t-sne		
Incremetal pca		
Feature agglo-meration		

	hepatitis using KMeans	
	PCA	t-SNE
Without feature reduction		
pca		
t-sne		
Incremetal pca		
Feature agglo-meration		

	satimage using KMeans	
	PCA	t-SNE
Without feature reduction		
pca		
t-sne		
Incremetal pca		
Feature agglo-meration		

	cmc using Agglomerative clustering	
	PCA	t-SNE
Without feature reduction		
PCA		
t-SNE		
Incremetal pca		
Feature agglo-meration		

	hepatitis Agglomerative clustering	
	PCA	t-SNE
Without feature reduction	**Clustering of hepatitis dataset using Agglomerative** PCA visualization without dimensionality reduction 	**Clustering of hepatitis dataset using Agglomerative** t-SNE visualization without dimensionality reduction 
PCA	**Clustering of hepatitis dataset using Agglomerative** PCA visualization With PCA for dim red to 12 features & 90.0 % exp. varianc 	**Clustering of hepatitis dataset using Agglomerative** t-SNE visualization With PCA for dim red to 12 features & 90.0 % exp. varianc 
t-SNE	**Clustering of hepatitis dataset using Agglomerative** PCA visualization With t-SNE for dim reduction to 12 features 	**Clustering of hepatitis dataset using Agglomerative** t-SNE visualization With t-SNE for dim reduction to 12 features 
Incremetal pca	**Clustering of hepatitis dataset using Agglomerative** PCA visualization with incremetal PCA for dim reduction to 12 features 	**Clustering of hepatitis dataset using Agglomerative** PCA visualization with incremetal PCA for dim reduction to 12 features 
Feature agglo-meration	**Clustering of hepatitis dataset using Agglomerative** PCA visualization With feature agglomeration for dim reduction to , 12 featur 	**Clustering of hepatitis dataset using Agglomerative** -SNE visualization With feature agglomeration for dim reduction to , 12 featu 

	Satimage Agglomerative clustering	
	PCA	t-SNE
Without feature reduction	**Clustering of satimage dataset using Agglomerative** PCA visualization without dimensionality reduction 	**Clustering of satimage dataset using Agglomerative** t-SNE visualization without dimensionality reduction 
pca	**Clustering of satimage dataset using Agglomerative** PCA visualization With PCA for dim red to 4 features & 90.0 % exp. variance 	**Clustering of satimage dataset using Agglomerative** t-SNE visualization With PCA for dim red to 4 features & 90.0 % exp. varianc 
t-sne	**Clustering of satimage dataset using Agglomerative** PCA visualization With t-SNE for dim reduction to 4 features 	**Clustering of satimage dataset using Agglomerative** t-SNE visualization With t-SNE for dim reduction to 4 features 
Incremetal pca	**Clustering of satimage dataset using Agglomerative** PCA visualization with incremetal PCA for dim reduction to 4 features 	**Clustering of satimage dataset using Agglomerative** PCA visualization with incremetal PCA for dim reduction to 4 features 
Feature agglo-meration	**Clustering of satimage dataset using Agglomerative** PCA visualization With feature agglomeration for dim reduction to , 4 feature 	**Clustering of satimage dataset using Agglomerative** t-SNE visualization With feature agglomeration for dim reduction to , 4 featur