CS7641 Assignment 3 – Unsupervised Learning

Nian Liu
Georgia Institute of Technology
nliu319@gatech.edu

I. Introduction

Unsupervised learning examines only the predictors of a dataset without being influenced by the responses (if they exist) to discover underlying patterns and structures within the data. It is commonly used in the context of exploratory data analysis to better understand the dataset or in combination with downstream supervised tasks to enable better performance or faster learning. In this project, we explore 2 specific classes of unsupervised learning methods – clustering and dimensionality reduction (DR). Clustering analyzes the similarities among the datapoints and assigns similar samples to the same cluster. It essentially "classifies" the datapoints by looking at the features along with a predefined similarity metric. Key to the success of clustering algorithms is how separated the various clusters are and how easy it is to draw the inter-cluster boundaries between them. While many algorithms exist to achieve this task, here we examine 2 commonly used methods, Gaussian Mixture Models (GMM) and K-means (KM), and make comparisons between them. By contrast, the principle of DR is fundamentally different in that it combines features from the original dataset to create a new feature set while retaining as much information as possible. The main advantage of this idea is that the transformed dataset frequently has fewer feature than the original, which generally benefits any machine learning algorithm due to the curse of dimensionality effect. The important considerations here are whether the original dataset has redundancy in the feature space to begin with, and whether the algorithm can find the appropriate projections that minimize loss of information. In particular, we will explore linear DR methods including Principal Component Analysis (PCA), Independent Component Analysis (ICA), and Random Projection (RP). A related non-linear manifold method, t-distributed Stochastic Neighbor Embedding (t-SNE), will also be used for the visualization of results in 2D space, although it will not be experimented with in detail. In the following sections, we will first introduce the 2 datasets selected for this project and why they can lead to interesting findings on the algorithms tested. We will then briefly describe the general workflow that applies to all sections of the project, with algorithm-specific workflows presented later in their respective sections. Afterwards, all experimental results and analyses will be presented. They are separated into 5 sections where we explore clustering and DR independently in the first 2 sections, analyze clustering in conjunction with DR in the 3rd section, and then examine how data preprocessing with either DR or clustering impacts supervised learning outcome in the remaining 2 sections. Finally, we will conclude this report with a summary of learnings and potential future work.

II. Description of Datasets

A. Dry bean

The dry bean dataset was originally published in [1], where 16 distinct geometric features that are all continuous were automatically extracted from dry bean images and used to classify 7 different types of beans. This dataset contains a total of 13,611 entries and generally leads to good classification performance when training supervised learning models (>0.92 test f1 score). Therefore, we hypothesize that the predictors in this dataset can be naturally grouped into distinct clusters that match their class labels even in a strictly unsupervised setting. We also performed a pair-wise correlation analysis on all 16 features (Fig. 1). Many of the features were highly correlated with each other, which is commonly the case when extracting multiple geometric features from the same image. Using a threshold of ± 0.8 to eliminate highly correlated features, this dataset can be effectively simplified to 6 uncorrelated features. Because of this, we also expect DR algorithms to be highly effective, where the number of features can be reduced significantly without loss of information. However, DR with ICA requires that the observed features are linear combinations of independent, non-Gaussian hidden variables [2]. Since dry beans are naturally occurring biological objects, it is reasonable to believe that the underlying hidden variables follow Gaussian distributions instead. Correspondingly, we hypothesize that ICA performance will be poor, with either an inability to find any independent components or the resulting projections having high reconstruction error.

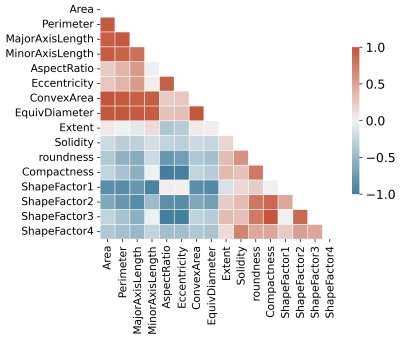

Fig. 1. Pair-wise correlation analysis on all features from the dry bean dataset.

B. Date fruit

The date fruit dataset [3] is highly similar to the dry bean dataset in that it uses the same geometric features extracted from images for the classification of 7 different types of dates. However, the key difference is that it also contains image color information, nearly doubling the numbers of features to 34 compared to that of dry bean. All features are also continuous here. Additionally, the dataset only contains 898 entries. We selected this dataset in conjunction with dry bean since it would be interesting to compare how the algorithms behave with many more features but significantly fewer examples to learn from, given all other data characteristics being highly similar. In this case, we hypothesize the clustering algorithms to perform much worse due to the curse of dimensionality effect. Nevertheless, due to the correlations observed in the feature set (Fig. 2), we still anticipate DR to be effective, except for ICA. Here, the number of uncorrelated features is 14 after applying the same ± 0.8 correlation threshold and analysis.

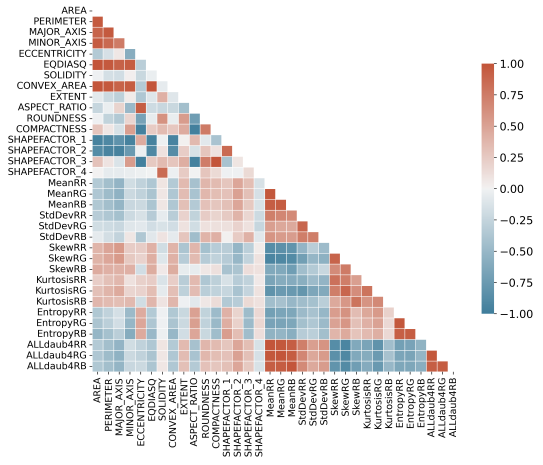

Fig. 2. Pair-wise correlation analysis on all features from the date fruit dataset.

III. General Workflow

For both datasets, a train-test split of 80:20 was performed using a stratified method based on class labels to ensure all splits had similar class distributions. The test set was used in the supervised learning sections for final performance evaluations only, and was not involved in any model tuning, training, or unsupervised learning tasks. For the training data, features were individually standardized by subtracting the mean and scaling to unit variance. The same scaling factors were then applied to the test set, such that we rely only on the training set for standardization. For the unsupervised learning sections, model hyperparameter tuning, such as determination of the number of clusters or components, was performed based on unsupervised metrics ignoring the class labels. Specific metrics used for each task will be detailed and justified below in the corresponding sections. The class labels were taken into consideration, however, when evaluating the results, especially for clustering. A similar idea was applied to the supervised learning sections, where we predetermined the clusters or transformed features prior to feeding them into a neural network (NN) for classification. This way, the various unsupervised methods were not biased by the class labels nor by the classification task itself. Supervised classification after unsupervised learning was only applied to the dry bean dataset, and the NN hyperparameters were chosen based on previous tuning efforts specifically on this dataset (1 hidden layer with 64 units; 0.1 dropout; SGD optimizer with 0.01 learning rate and 0.9 momentum).

IV. Results

A. Unsupervised clustering

Clustering algorithms group datapoints based on their feature values such that intra-cluster points are more similar to each other than inter-cluster points. GMM and KM were the 2 algorithms explored here. We chose the Calinski–Harabasz score (CH score) as the metric for performance evaluation for 2 reasons: (1) it is an unsupervised metric; (2) it accounts for both intra-cluster similarity and inter-cluster difference. A high CH score indicates good clustering results as it requires both the inter-cluster distance to be high and the intra-cluster distance to be low. Implicit in its definition is that the CH score uses Euclidean distance as a measure of similarity. Both of our datasets contain only continuous features that can be meaningfully described in the Cartesian space, and we performed z-standardization such that values across features are on the same scale. Therefore, the Euclidean distance is suitable. Additionally for GMM, results may depend on the type of covariance matrix used, where a "full" matrix allows for any cluster shape, "tied" forces all clusters to have the same shape (can be elliptical), "diagonal" forces all cluster axes to be orthogonal to each other (but otherwise can be any shape), and "spherical" forces each cluster to have equal variances in all dimensions. In practice, the choice of covariance matrix did not significantly affect the results shown in Fig. 3, with the exception of "tied". Our z-standardization method likely caused variances across feature dimensions to be similar within each cluster. Correspondingly, all covariance matrix types can be effectively reduced to the "spherical" case, explaining the similarities. The different "tied" result is likely due to the differences in inter-cluster variance, meaning that the various clusters have different sphere diameters. Nevertheless, we opted to use the full covariance matrix for all subsequent analysis since other manipulations may affect the shape of the clusters making the spherical assumption no longer valid. To choose the optimal k value, we observed the number of clusters that led to the highest CH score. It is clear that for the date fruit dataset, $k = 3$ is optimal. However for the dry bean dataset, we observed 2 peaks in the score curves, $k = 2$ or $k = 5$, across both GMM and KM results. To differentiate between the two, we also examined the distribution of Silhouette scores across all datapoints. Similar to the CH score, Silhouette scores consider both intra- and inter-cluster distances but can be defined for each individual sample. The results shown in Fig. 4 indicates that $k = 2$ (left panel) is not a good choice

given that nearly all points within cluster 0 had below average scores, and a small portion of datapoints did not belong to that cluster (negative Silhouette scores). Therefore, $k = 5$ (middle panel) was chosen for the dry bean dataset. There were no issues with the score distribution for the date fruit dataset at $k = 3$ (Fig. 4 right).

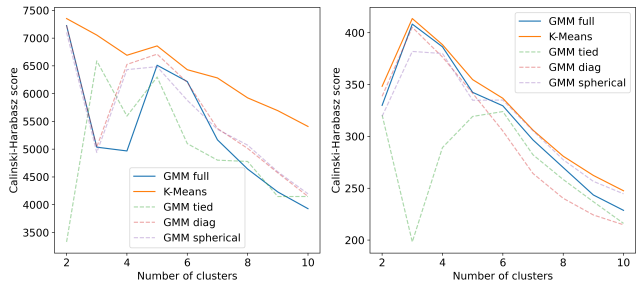

Fig. 3. CH score as a function of the number of clusters for GMM and KM. Results for the various GMM covariance matrix types are also shown. **Left**: results for the dry bean dataset; **right**: results for the date fruit dataset.

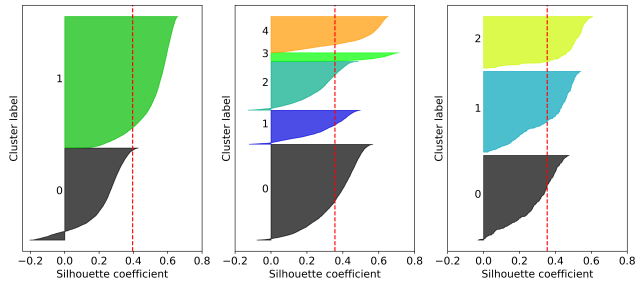

Fig. 4. Distribution of individual Silhouette scores aggregated by their assigned clusters. Results were generated using the KM model (GMM showed similar trends). **Left**, dry bean dataset when setting $k = 2$; **middle**, dry bean dataset when setting $k = 5$; **right**, date fruit dataset when setting $k = 3$. The dotted vertical line indicates the average Silhouette scores across all datapoints.

To further understand the results, we visualized the high dimensional datasets along with the assigned clusters on 2D plots after projection with t-SNE, as shown in Fig. 5. Note that the t-SNE projection was performed in an unsupervised manner without any input from cluster or class labels. Regardless of the algorithm used, the clustering results align with the projection groupings fairly well. Islands in the data were assigned consistent clusters and clear inter-cluster boundaries can be seen in other cases. Comparing the 2 datasets however, there are significantly fewer groupings for date fruit dataset indicated by t-SNE projections, which corresponds to the lower optimal k value. This is consistent with our original hypothesis. The date fruit dataset has almost twice the amount of features with less than a tenth of examples, which leads to a higher degree of intra-cluster variance. This in turn makes it challenging for any algorithm to define the inter-cluster boundary, forcing multiple groupings to be lumped together. Evaluating clusters in conjunction with class labels reveals similar findings (Fig. 6 and Table I). For both datasets, the number of clusters is fewer than the number of classes, indicating that clusters indeed contain multiple classes which lead to low homogeneity scores. Once again, the homogeneity score for the date fruit dataset is lower than that of dry bean. Taken together, these results demonstrate that unsupervised learning also suffers from the curse of dimensionality effect. In a sense, it is impacted more than supervised learning since it relies only on distributions of predictors without information provided by class labels to help delineate between noise and true differences. It was also interesting to see some of the clustering results align well with the classification results from Assignment 1. For example as shown in Fig. 6, both algorithms cluster classes 3 and 6 together, and misclassifications between classes 3 and 6 were also frequently observed in the confusion matrices across all supervised algorithms tested [4]. In these situations, the performance of algorithms is limited by the data characteristics and collecting additional features might be a better solution to booster accuracy. Lastly, comparing between the 2 clustering algorithms, both offer very similar performance across all metrics analyzed. As mentioned previously, the datasets describe naturally occurring objects where feature values are expected to be normally distributed (see Assignment 1 for confirmation [4]). After z-standardization, the dataset becomes very well-behaved with minimal outliers, and the classes that can be separated each fall into roughly spherical shapes with distinct boundaries (e.g. the islands and natural groupings shown in the t-SNE projections). In these cases, the benefits from GMM soft clustering, such as robustness to outliers and better modeling of complex boundaries, are no longer advantageous. In other words, the simpler KM model already has a sufficient hypothesis space to capture the structures of both of our datasets. This lower complexity of KM leads to faster training times, demanding only 169/109 ms (dry bean/date fruit) compared to 16,347/2,250 ms for GMM.

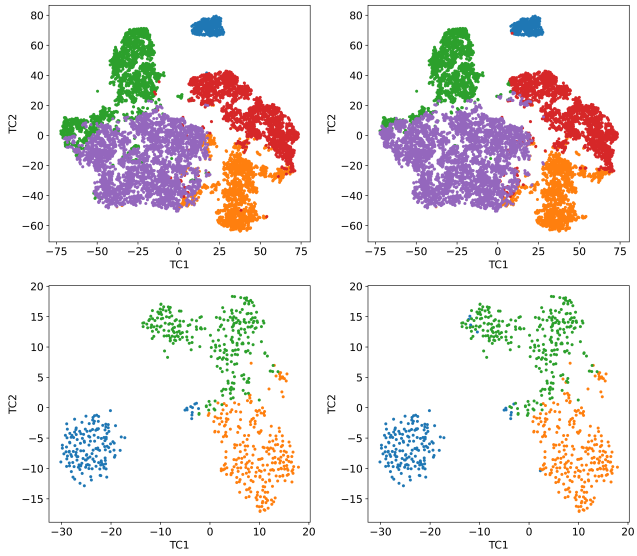

Fig. 5. Visualization of clustering results on the dry bean dataset ($k = 5$) and date fruit dataset ($k = 3$). **Top row**: dry bean dataset; **bottom row**: date fruit dataset. **Left column**: GMM results; **right column**: KM results. The 2D plots were created by projecting the original data using t-SNE.

TABLE I
CLUSTERING HOMOGENEITY AND COMPLETENESS SCORES

Dataset	DR	Clustering	Homogeneity	Completeness
Dry bean	None	GMM	0.597	0.754
Dry bean	None	KM	0.610	0.799
Dry bean	PCA	GMM	0.560	0.732
Dry bean	PCA	KM	0.608	0.794
Dry bean	ICA	GMM	0.458	0.516
Dry bean	ICA	KM	0.502	0.657
Dry bean	RP	GMM	0.541	0.699
Dry bean	RP	KM	0.581	0.798
Date fruit	None	GMM	0.501	0.864
Date fruit	None	KM	0.486	0.837
Date fruit	PCA	GMM	0.602	0.838
Date fruit	PCA	KM	0.485	0.834
Date fruit	ICA	GMM	0.353	0.736
Date fruit	ICA	KM	0.549	0.742
Date fruit	RP	GMM	0.503	0.874
Date fruit	RP	KM	0.460	0.794

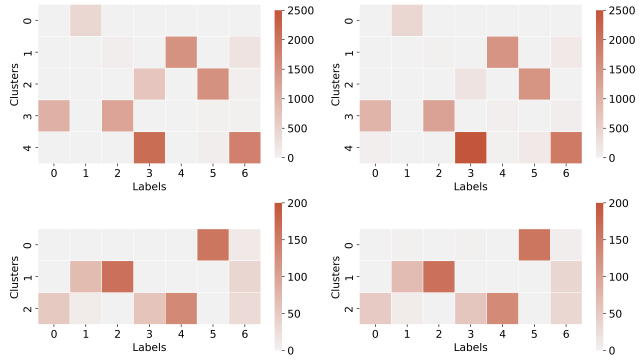

Fig. 6. Heatmaps comparing the datapoints' assigned cluster labels versus the ground truth class labels. **Top row**: dry bean dataset; **bottom row**: date fruit dataset. **Left column**: GMM results; **right column**: KM results. The specific ordering of assignments does not necessarily match between the cluster labels and class labels.

B. Unsupervised dimensionality reduction

For this section, the purpose of DR is to find a subset of new features which are linear combinations of the original features such that the projected dataset has fewer dimensions while still retaining as much information from the original dataset as possible. Here we explored PCA, ICA, and RP, which all have distinct mechanisms for data projection. PCA attempts to find a new set of directions that are both uncorrelated and orthogonal to each other while explaining as much of the dataset's variance as possible. To determine the optimal number of new features, we examined the cumulative explained variance and error after data projection then reconstruction. We previously analyzed that both datasets have many redundant features due to multicollinearity, making PCA a particularly suitable algorithm. As shown in Fig. 7, we were able to effectively reduce the dimension of the dry bean dataset from the original 16 to 5, and from 34 to 8 with the date fruit dataset. These values were chosen such that the top components (e.g. those with the highest eigenvalues) collectively accounted for $\geq$95% of the cumulative explained variance. With these PCs, the reconstruction error is also low, suggesting that the projection to a lower dimension space indeed preserves most of the information. Interestingly, PCA is more effective with the date fruit dataset even though it has fewer examples. This suggests that it does not rely on as many training examples to perform well compared to the clustering algorithms examined above. One possible explanation is that PCA only needs to analyze data along each dimension, whereas clustering and classification algorithms require the entire multidimensional space to be characterized, which is much more challenging. While the optimal number of PCs is very similar to the number of uncorrelated features from the correlation analyses in section II, a deeper analysis of the PCs indicates that those with the highest eigenvalues integrate information from all original features with nearly equal absolute loading values (Fig. 8). This suggests that although many features are highly correlated, each of them still contain distinctive information about the dataset. Therefore, they are not necessarily redundant, and feature elimination based on correlation alone could lead to a loss of information.

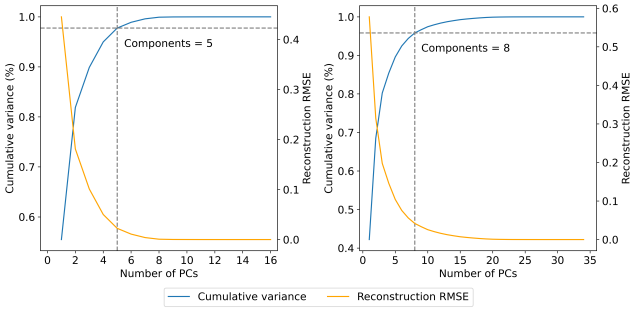

Fig. 7. Cumulative explained variance and reconstruction error after PCA with varying numbers of PCs. **Left**: results for the dry bean dataset; **right**: results for the date fruit dataset. For DR purposes, the optimal components were chosen such that they collectively accounted for >95% of the cumulative explained variance.

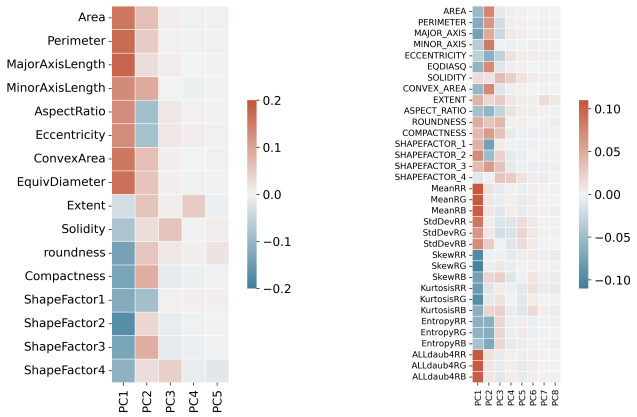

Fig. 8. Visualization of the contribution of each feature towards each PC (e.g. eigenvector values). Contributions are normalized by each PC's explained variance. **Left**: results for the dry bean dataset; **right**: results for the date fruit dataset.

ICA finds a new set of independent dimensions from

the original features while maximizing mutual information between the projected and unaltered data. To find ICs, the algorithm relies on nongaussianity along the dimensions of the transformed space, with higher nongaussianity indicating better independence [2]. As such, we used kurtosis as a key measure for IC selection and, similar to PCA, we chose the top components that collectively accounted for $\geq$95% of the total kurtosis. This resulted in 8 and 17 components for the dry bean and date fruit datasets, respectively (Fig. 9). The results clearly show a hierarchy in terms of the nongaussianity of components, with the first 2 components having significantly higher kurtosis than the rest for both cases. This illustrates that the datasets can be projected into independent features and DR is possible on a surface level. However, the reconstruction error shows a linear trend with respect to the number of ICs, which suggests that information from the original dataset is evenly distributed among the components. This is in contrast to PCA where the top few PCs can already achieve near perfect reconstruction. Correspondingly, we conclude that using ICA for DR on either dataset is ineffective. Fig. 10 confirms these observations in that each IC is a linear combination of only a few of the original features. Looking across all ICs, many of the features do not contribute anything towards projection and the information they contain is lost during transformation. This is likely a direct consequence of ICA striving to find independent components, forcing each IC to only combine limited features due to their interdependency (i.e. some features can be derived from others). Therefore, each IC only contains a small amount of information of the original data, and nearly all components are needed to match the reconstruction error of PCA. Another potential reason follows our previous argument that the hidden variables are likely to be Gaussian distributed due to them belonging to a naturally occurring biological process. This violates the fundamental assumption of ICA, making it very difficult to find the proper set of independent features, leading to poor performance.

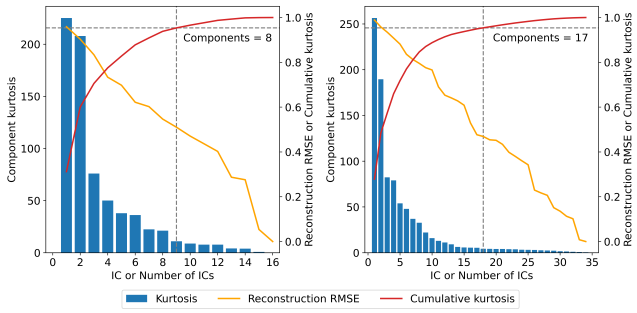

Fig. 9. The kurtosis values of each individual IC after ICA. Cumulative kurtosis as a fraction of total kurtosis and reconstruction error with varying numbers of ICs are also shown. **Left**: results for the dry bean dataset; **right**: results for the date fruit dataset. For DR purposes, the optimal components were chosen such that they collectively accounted for >95% of the cumulative kurtosis.

Unlike PCA or ICA which performs projections according to specific objective functions, RP projects randomly. RP becomes advantageous when it is applied to data with

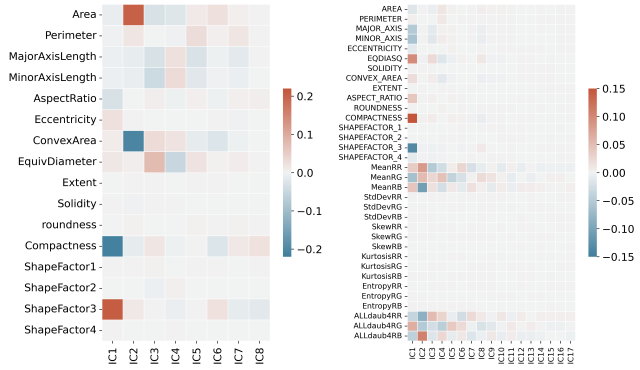

Fig. 10. Visualization of the contribution of each feature towards each IC. Contributions are normalized by each IC's kurtosis. **Left**: results for the dry bean dataset; **right**: results for the date fruit dataset.

hundreds to thousands of features. In these cases, RP can very rapidly project the original data into a lower dimensional space with the Johnson–Lindenstrauss lemma guaranteeing minimal changes to all pair-wise distances of datapoints. The datasets we chose do not have sufficiently high feature dimensions and hence RP may not work very well. Fig. 11 shows the mean and standard deviation of the reconstruction error as functions of the number of projection components over 5 randomized runs. The small error regions indicate that RP results are reproducible and can meaningfully transform the original data into a lower dimension space. However, it suffers from a similar issue as ICA where the reconstruction error decreases linearly with increasing components. This also means that DR using RP is ineffective in that while reduction of the feature space is possible, the amount of information lost during the process is proportional to the decrease in dimension. This suggests that in addition to each original feature carrying information about the dataset, very specific combinations are required to relay that information, and PCA is the only algorithm examined that can do so. Additionally, our dataset and feature sizes are simply too small to take advantage of the fast computational speeds of RP as it only slightly outcompetes PCA in runtime (computational times expressed in ms for dry bean/date fruit: RP – 0.27/0.24; PCA – 0.47/0.40; ICA – 238/20). Overall, RP is not suitable for either of our datasets. For the purposes of later sections, we still transformed the datasets using 6 and 14 random components, respectively, which were chosen based on previous correlation analysis. This was done because there were no clear indications on the optimal component number based on reconstruction error alone.

C. Unsupervised clustering after dimensionality reduction

Using the DR algorithms and their respective optimal number of components for projection, we transformed the dry bean and date fruit datasets resulting in 6 new sets of data. We then applied GMM and KM clustering algorithms on the projected datasets and analyzed their results. We followed a similar strategy outlined in section IV-A where we chose the k value based on the unsupervised CH score (Fig. 12).

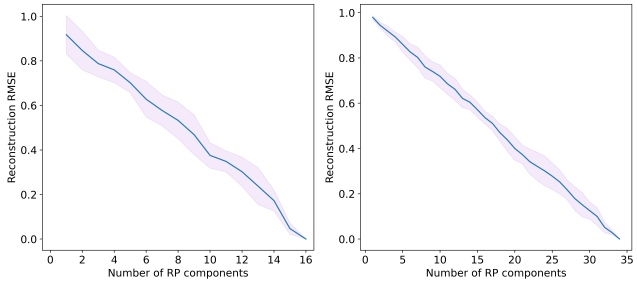

Fig. 11. Reconstruction error after RP with varying numbers of randomly projected components. **Left**: results for the dry bean dataset; **right**: results for the date fruit dataset. The mean and standard deviation across 5 random runs are shown.

Then for cluster evaluation, we calculated the homogeneity and completeness scores considering the class labels (Table I), while also showing a heatmap visualization of the result (Fig. 13). As discussed previously, we only focused on the full covariance matrix for GMM since it provided the best flexibility in terms of capturing any potential changes to cluster shapes. For the dry bean dataset, whereas additional analyses were needed to choose between $k = 2$ and $k = 5$ for the clustering-only experiments, it is obvious from the CH scores alone that $k = 5$ is optimal when clustering after PCA DR. An additional peak at $k = 7$ also appears, which matches the number of classes in the dataset. These results indicate that our algorithms are able to learn more effectively once the feature dimension is reduced. A similar phenomenon was observed with the date fruit dataset, where the optimal number of clusters increased from 3 to 4 when clustering with GMM after PCA DR, bringing it closer to the number of ground truth classes. Consequently, the homogeneity score significantly increased without sacrificing completeness (Table I), and some of the classes were better separated into their own clusters (Fig. 13 compared to Fig. 6). Overall, results from both datasets point towards the benefit of performing DR with PCA on clustering, further confirming our previous analysis on the curse of dimensionality. Surprisingly though, KM did not seem to benefit as much. One explanation is that PCA DR helps reveal additional intricacies in the cluster boundaries that can only be captured by the soft clustering nature and better cluster shape modeling capabilities of GMM (since the full covariance matrix was used). Indeed, a visualization of the clusters shown in Fig. 14 confirms that GMM, but not KM, was able to split a large cluster into 2 new clusters after PCA DR without losing resolution on any of the other clusters.

In contrast to PCA, DR with ICA resulted in worse clustering performance. Regardless of the algorithm and dataset, the CH score dropped significantly, indicating poor intra-cluster consistency, inter-cluster separation, or both. There are no longer clear separation of classes into clusters (Fig. 13), negatively impacting both homogeneity and completeness (Table I). We previously analyzed that ICA transformation discarded a lot of information from the original features, which likely explains the poor performance seen here. Apart from

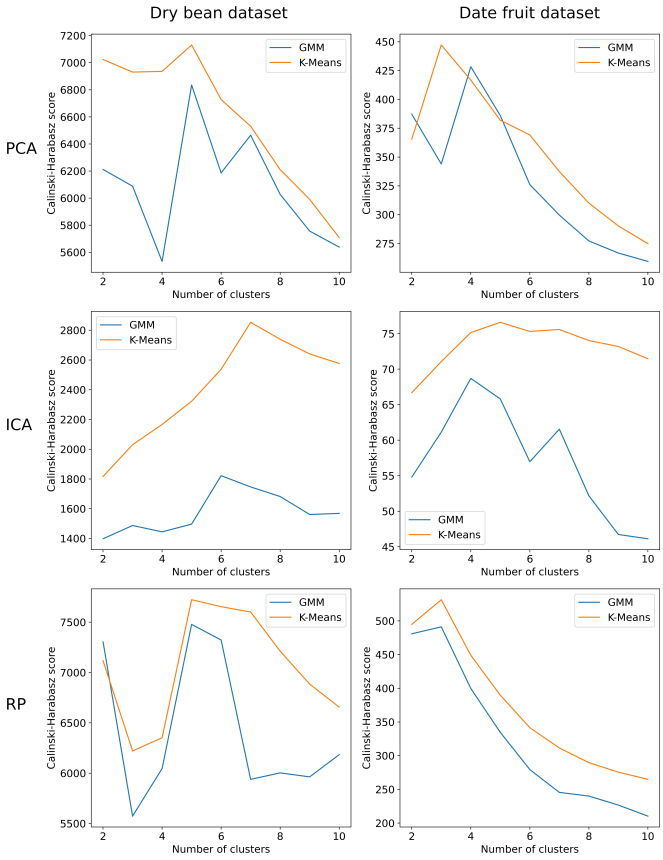

Fig. 12. CH score as a function of the number of clusters for GMM and KM after DR with PCA, ICA, or RP was applied on the original datasets. The full covariance matrix was used for GMM. **Left column**: results for the dry bean dataset; **right column**: results for the date fruit dataset. **Top row**: clustering after PCA; **middle row**: clustering after ICA; **bottom row**: clustering after RP.

that, an interesting observation is that ICA seemed to impact GMM more than KM (much lower CH scores for GMM). This is likely because ICA transforms the original features into those with high nongaussianity, whereas GMM expects the data to be clustered into Gaussian regions. Therefore, these 2 algorithms are fundamentally not compatible with each other. Overall, the clustering results after ICA projection on either dataset were not meaningful.

Finally, clustering results can be maintained after applying DR with RP. The optimal number of clusters remained unchanged (Fig. 12), and even the aggregation of ground truth classes into clusters remained largely unchanged (Fig. 13 and Table I). Our explanation here is that while RP has likely changed the pair-wise distances throughout the dataset (dimensions well below the Johnson–Lindenstrauss bound) and some information is lost in the process, the clusters remain distinctive enough such that boundaries can still be drawn. We fully anticipate that for more refined tasks, such as classification of all 7 classes, projecting the data with RP will negatively impact performance.

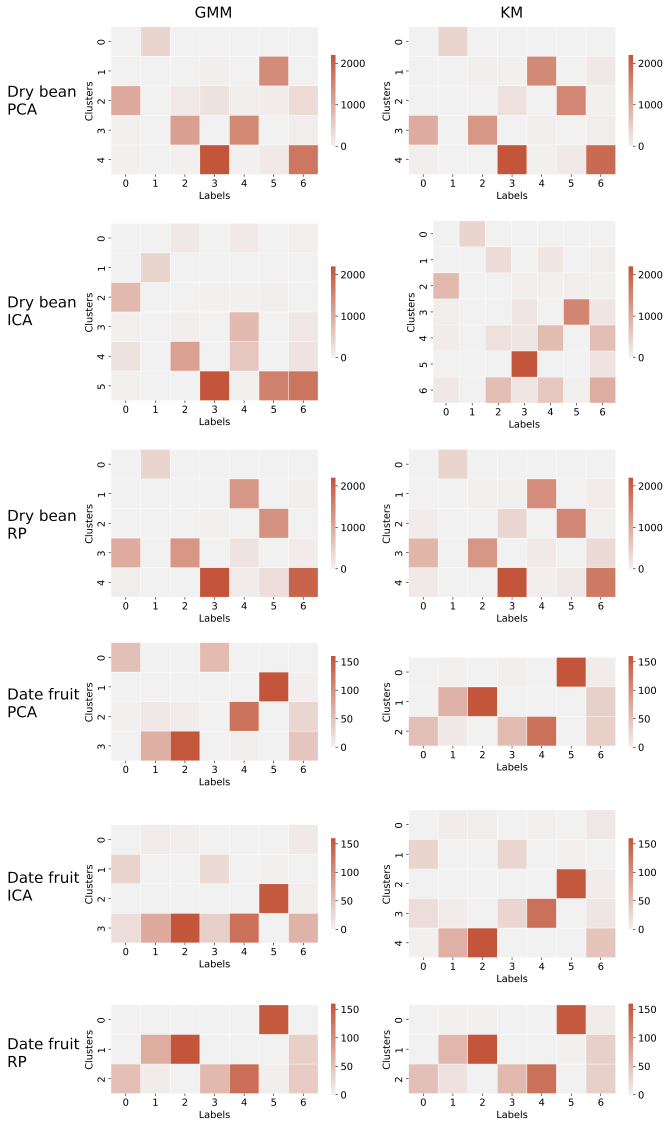

Fig. 13. Heatmaps comparing the datapoints' assigned cluster labels using either GMM or KM versus the ground truth class labels for the dry bean and the date fruit datasets. DR with PCA, ICA, or RP was applied prior to clustering. **Row 1**: dry bean dataset after PCA; **Row 2**: dry bean dataset after ICA; **Row 3**: dry bean dataset after RP; **Row 4**: date fruit dataset after PCA; **Row 5**: date fruit dataset after ICA; **Row 6**: date fruit dataset after RP. **Left column**: GMM results; **right column**: KM results. The specific ordering of assignments does not necessarily match between the cluster labels and class labels.

D. Supervised learning with dimensionality reduction

To understand how DR impacts supervised classification results, we chose the PCA, ICA, or RP projected dry bean datasets, trained NN classifiers, and compared their results along with that of the original dataset. DR was run prior to and independent of NN training and hence the same number of projection components as those determined in section IV-B was used here. Following our previous analysis, we expect PCA transformation to provide the most benefit, with ICA and RP potentially having negative impacts due to significant loss of information. Fig. 15 shows the 5-fold cross validation

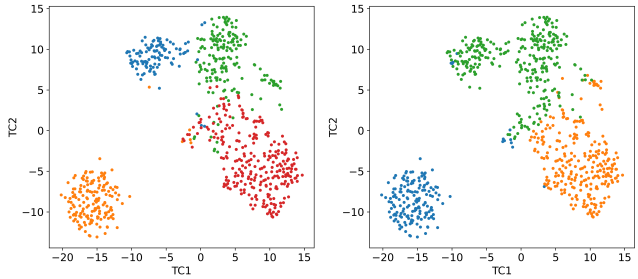

Fig. 14. Visualization of clustering results on the date fruit dataset after PCA projection. **Left**: GMM with optimal $k = 4$; **right**: KM with optimal $k = 3$. The 2D plots were created by projecting the original data using t-SNE.

f1 score as functions of either the training epoch or training size. F1 score was chosen as the metric since it is suitable for imbalanced datasets (such as dry bean) by accounting for false positives and false negatives. There were no performance gains even after applying PCA DR. This means that when trained on the original dataset with many redundant features, the NN was able to learn and distill useful information while ignoring those that did not contribute to classification. ICA and RP did negatively impact both validation (Fig. 15) and test f1 scores (Table II) due to the same reasons mentioned in the previous sections. In general, performing DR does not add any new information to the dataset, but rather extracts useful information ahead of time such that the network can spend less time determining the relevance of features. This explains why convergence is faster when any DR was applied, compared to the control (Fig. 15 and Table II). However, reducing the number of features by more than half (16 to 5 – 8 depending on the algorithm) did not have an appreciable impact on the training time per epoch. In terms of training weights, reductions in feature dimensionality only affected the input layer, while the majority of weights came from the hidden layer. For example, the NN applied to the PCA transformed data had 5,063 weights compared to the baseline case of 5,703. The reduction of trainable weights was small, explaining the minimal impact on training time. Lastly, the learning curves shown in the right panel of Fig. 15 shows that the classifier was able to learn more effectively at smaller training sizes if the dataset is transformed into fewer features. This perfectly highlights the curse of dimensionality once again and why PCA DR is a useful tool for data preprocessing ahead of supervised learning.

E. Supervised learning with cluster augmentation

To understand whether clustering can also impact classification results, we took the cluster assignments obtained in section IV-A, performed one-hot encoding, and appended them to the original dry bean dataset as extra features. Adding the extra features only slightly increased training time as changes to the overall number of trainable weights remain low (Table II). However, they did not provide any extra information to the classifier since the validation and test f1 scores were nearly identical to the baseline (Fig. 16 and Table II). The epochs

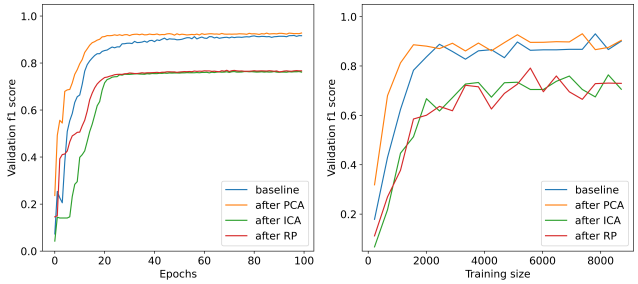

Fig. 15. Comparison of NN classification results when using the original dry bean dataset, as well as PCA, ICA, or RP transformed version of the same dataset. **Left**: 5-fold cross validation f1 scores as functions of training epochs; **right**: 5-fold cross validation f1 scores as functions of training data sizes.

TABLE II
DRY BEAN CLASSIFICATION RESULTS

Preprocessing	Train time[a]	Convergence[b]	Test f1 score
None	124	42	0.930
w/ PCA DR	116	34	0.934
w/ ICA DR	119	38	0.767
w/ RP DR	117	37	0.764
w/ GMM cluster	127	44	0.931
w/ KM cluster	128	43	0.933

[a]Runtime in ms per epoch.
[b]Epochs required for less than 0.001 change in validation loss averaged over a 15 epoch window.

required for convergence also slightly increased because the network needed to learn that the extra features did not provide anything new. This result is expected. The clustering labels were derived directly from the original data so we do not expect them to contain additional information. In some cases, they facilitate classifier learning by providing some information about how the predictors are distributed ahead of time. However in our case, the cluster homogeneity scores were low, which means that the network would still need to rely on the original data in order to learn how to differentiate among classes within any cluster. To confirm this hypothesis, we also trained the network with the clustering labels alone, and unsurprisingly, performance was poor with only 0.750 or 0.757 validation f1 score using GMM or KM clusters, respectively. This shows that the information contained in the cluster labels is incomplete from a classification perspective, which limits their usefulness and renders them redundant to the original dataset. Comparing DR versus clustering as 2 data preprocessing methods prior to supervised learning, neither provides any new information than what the dataset already contains. Rather, both aim to distill relevant information making it easier or faster for the classifier to learn. For the dry bean dataset specifically, distilling information is easy to begin with, and hence the comparison should focus on which preprocessing algorithms can transform the data to the minimal possible size while still retaining complete information. Based on all of our analyses in the unsupervised learning sections, the only algorithm that can achieve both of these requirements is PCA, which is in turn reflected by the improvements seen in terms of convergence rate and the amount of data required to achieve good learning.

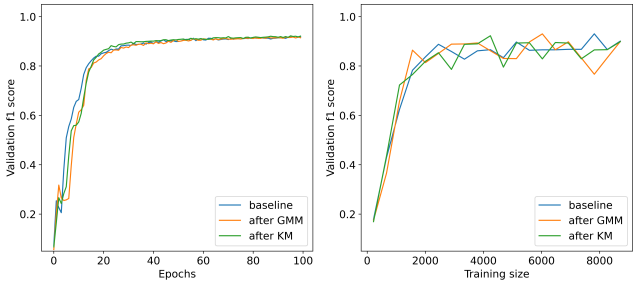

Fig. 16. Comparison of NN classification results when applied to the original dry bean dataset, as well as when the dataset is augmented with cluster labels from GMM or KM clustering. The cluster labels are one-hot encoded and then appended to the dataset as extra features. **Left**: 5-fold cross validation f1 scores as functions of training epochs; **right**: 5-fold cross validation f1 scores as functions of training data sizes.

V. LEARNINGS AND FUTURE WORK

In this project, we explored various unsupervised clustering and DR algorithms, analyzed and compared their differences, and applied them in the context of preprocessing for supervised classification. We specifically chose datasets that have a high degree of feature redundancy and analyzed how the methods performed when subjected to increasing feature dimensions while having fewer training examples. Our key finding is that feature dimensionality greatly impacts both unsupervised and supervised learning, due to the curse of dimensionality. This is evident in comparisons across the 2 datasets as well as in comparisons before and after DR on the same dataset. In addition, we learned that using clustering or DR as feature preprocessing techniques can generally benefit machine learning tasks as long as they are able to (1) distill useful information from the original dataset while removing redundancies and (2) retain complete information from the original dataset.

For future work, it would be interesting to explore different datasets to understand whether preprocessing with clustering or DR can lead to meaningful improvements to the actual performance of classifiers. While clustering and DR to not add new information, there may be cases where information crucial to classification is too confounded in the dataset and difficult for supervised learning methods to uncover. In these cases, unsupervised learning methods might better present the information through either feature transformation or added features, leading to better classification accuracy.

REFERENCES

[1] M. Koklu and I. Ali, "Multiclass classification of dry beans using computer vision and machine learning techniques," Computers and Electronics in Agriculture, vol. 174, 2020.
[2] A. Hyvärinen and E. Oja, "Independent Component Analysis: Algorithms and Applications," Neural Networks, vol. 13, pp. 411-430, 2000.
[3] M. Koklu, R. Kursun, Y.S. Taspinar, and I. Cinar, "Classification of Date Fruits into Genetic Varieties Using Image Analysis," Mathematical Problems in Engineering, 2021.
[4] N. Liu, "CS7641 Assignment 1 – Supervised Learning," 2024