Assessment of Clustering and Dimensionality Reduction Algorithms

Andreas Koutras

I. Introduction

This project evaluates the effectiveness of different clustering and dimensionality reduction algorithms. Specifically, it examines K-Means and Expectation Maximization (EM) for clustering, and Principal Component Analysis (PCA), Independent Component Analysis (ICA), and Randomized Projections (RP) for dimensionality reduction and explores the potential of these techniques to improve the performance of a neural network. The analysis is conducted on two datasets from the UCI Machine Learning Repository: the *Statlog Landsat Satellite* dataset [1] and the *Default of Credit Card Clients* dataset [2], hereafter referred to as the Landsat and Default datasets, respectively. These datasets were chosen for their contrasting characteristics: the Landsat dataset has six target classes and exclusively numerical features, while the Default dataset has a binary target class and a mix of numerical and categorical features. These differences enable a comparative study of algorithm performance across varied data types and target structures. All the analyses were conducted using the scikit-learn [3] library in Python.

II. Data Sets

The Landsat dataset has 6,435 features and contains 36 numerical features, each ranging from 0 to 255, representing multi-spectral pixel values in 3x3 neighborhoods within a satellite image of the Earth's surface. The target variable has six distinct classes — 1, 2, 3, 4, 5, and 7 (with no class 6) — each corresponding to a specific type of land cover, such as red soil (Class 1), cotton crop (Class 2), etc. The distribution of classes within the dataset is 24%, 11%, 21%, 10%, 11%, and 23% for the six classes respectively, indicating a class imbalance with a maximum ratio of 1:2.4. The Default dataset has 29,965 samples and contains 23 features of mixed types. These include one binary feature (gender), two categorical features (education level and marital status), and six discrete integer features representing payment delays in months (ranging from 1 to 9). The remaining features are integer values indicating dollar amounts for credit limits, bill statements, and previous payments.Unlike the Landsat dataset, the Default dataset has a binary target variable, where a value of 1 indicates a client has defaulted, and 0 indicates no default. The class distribution is notably imbalanced: 78% of samples are in class 0 (no default), while 22% are in class 1 (default). The marital status feature was converted to binary features using one-hot-encoding resulting in a total of 25 features for the Default dataset.

The dataset was shuffled and split into a training set (70%) and a test set (30%). To address the sensitivity of clustering algorithms to feature scaling, the training set was standardized, and the same transformation was applied to the test set. Clustering and dimensionality reduction algorithms were fitted exclusively on the training data, while the test data was reserved for evaluating the performance of the neural network.

To visualize class distribution in the datasets, we generated the three-dimensional scatter plots shown in Fig. 1. The axes represent the three most important features, determined by Gini impurity reduction in a decision tree model. In Landsat some classes form cohesive clusters whereas in the Default set class cluster are not as well defined. Therefore, we expect that the clustering algorithms will perform better in the Landsat dataset over the Default dataset. Fig. 2 presents the two-dimensional t-SNE representations of the datasets. While t-SNE effectively captures some clusters in the Landsat dataset, it fails to separate the two classes in the Default dataset, further highlighting the complexity and ambiguity of the latter. The next section assesses the performance of K-Means and EM for clustering.

III. Unsupervised clustering

In this section, we selected the number of clusters for K-Means and EM in an unsupervised manner, assuming that the ground truth is not known. We then evaluated the clustering performance by comparing the results to the ground truth. Only the training set was used for this analysis.

A. K-Means

To evaluate the K-Means clustering quality and select an appropriate number of clusters, we used three metrics: inertia, silhouette score, and Calinski-Harabasz (CH) score. Inertia measures the compactness of clusters by calculating the sum of squared distances between points and their cluster centroid. Lower values indicate tighter clusters. Silhouette score assesses separation between clusters, ranging from -1 to 1 [3]. Higher values mean well-defined, distinct clusters. Th CH score captures both compactness and separation [3]. Higher scores indicate more distinct and well-separated clusters. Each metric provides a unique view of clustering quality, highlighting different aspects of K-Means performance.

Fig. 3 and Fig. 4 show how the inertia, Calinski-Harabasz (CH), and silhouette score vary with the number of clusters for the Landsat dataset. The inertia curve reveals notable changes in the rate of decrease at 3 and 6 clusters suggesting an elbow point around these values. Both the average silhouette score

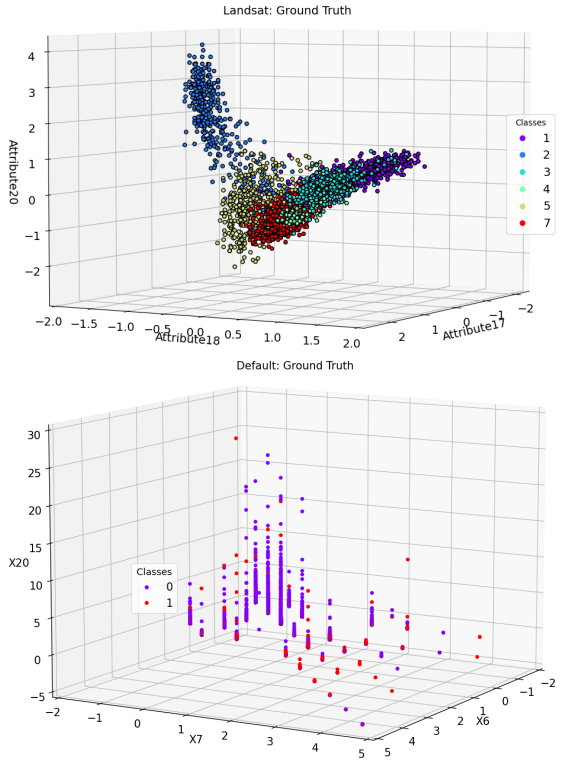

Fig. 1: Ground truth class scatter plots for the Landsat (top) and Default (bottom) datasets.

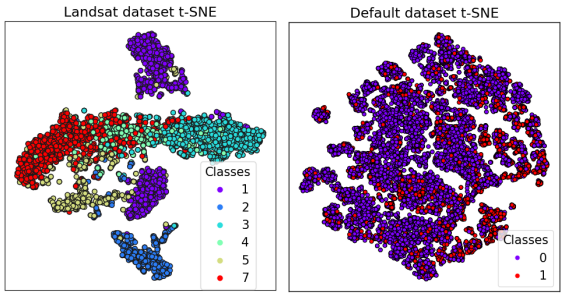

Fig. 2: Two-dimensional t-SNE representations of the datasets using a perplexity parameter of 50.

and CH score peak at 3 clusters but remain relatively high from 3 to 6 clusters. To assist in selecting the optimal cluster count, we plotted the silhouette coefficient of each data point within each cluster (Fig. 4), shown here for the 4-cluster case. This analysis confirmed that the clusters are reasonably sized, without being overly narrow or wide, and that all clusters meet or exceed the average silhouette score. Based on these criteria, we concluded that 3, 4, or 6 clusters are viable choices, with 4 clusters selected as a balanced option.

Fig. 5 and Fig. 6 display the clustering metrics for the Default dataset. The inertia curve does not reveal a clear elbow point. However, the silhouette score suggests that two clusters would be reasonable, while the CH score indicates that either two or three clusters could be appropriate. After examining the silhouette plots for both options, we concluded that three

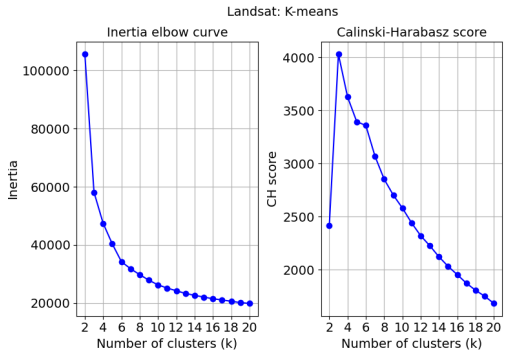

Fig. 3: K-means clustering of the Landsat dataset: inertia and Calinski-Harabasz with respect to number of clusters.

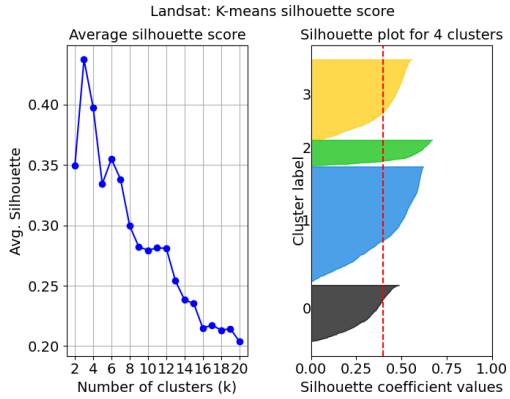

Fig. 4: K-means clustering of the Landsat dataset: average silhouette score with respect to number of clusters and silhouette plot for 4 clusters.

clusters are a better choice, as the two-cluster solution showed a significant discrepancy in silhouette scores between the clusters.

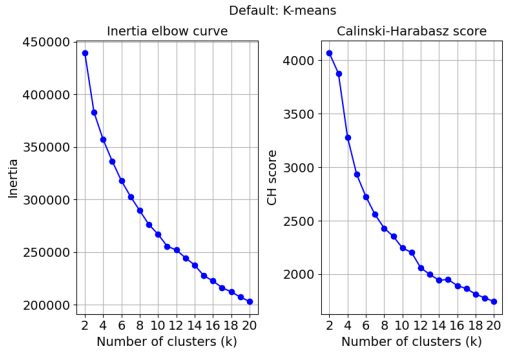

Fig. 5: K-means clustering of the Default dataset: inertia and Calinski-Harabasz with respect to number of clusters.

We then compared the unsupervised clustering results with the ground truth. Fig. 7 and Fig. 8 show scatter plots of the clusters generated for the Landsat and Default datasets, along with bar plots indicating the proportion of data points from each ground truth class assigned to each cluster. The

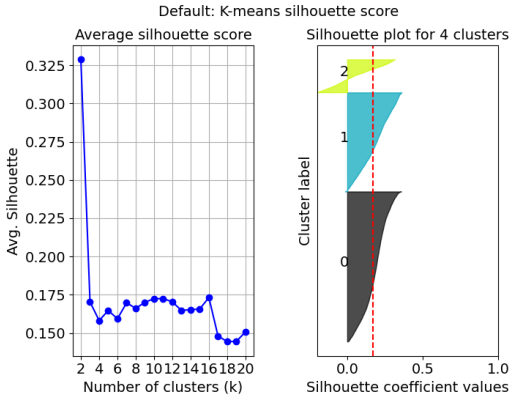

Fig. 6: K-means clustering of the Default dataset: average silhouette score with respect to number of clusters and silhouette plot for 3 clusters.

results confirm our hypothesis that clustering would be more straightforward for the Landsat dataset. In this case, Class 2, which is clearly separated from the rest, was almost entirely assigned to a single cluster, Cluster 2, composed exclusively of data points from Class 2. Cluster 1 primarily captured Class 1, while the remaining three classes, positioned near the center of the scatter plot, were distributed across the remaining two clusters. In the Default dataset, each of the three clusters contains data points from both classes, indicating that k-means did not identify a meaningful clustering structure. A similar pattern was observed when using two clusters. To quantify clustering performance against the ground truth, we used the V-measure score, which is the harmonic mean of homogeneity and completeness. A V-measure of 1.0 indicates perfect clustering. The resulting V-measure was 0.56 for Landsat and just 0.01 for the Default dataset.

B. Expectation Maximization

The Expectation-Maximization (EM) method was applied to a Gaussian Mixture Model (GMM). To determine the optimal number of clusters, we used the Bayesian Information Criterion (BIC), which is well-suited to the probabilistic nature of GMMs. Derived from Bayesian principles, BIC aligns with the likelihood maximization in GMMs and effectively quantifies the trade-off between model fit and complexity. The optimal GMM corresponds to the model with the smallest BIC, including negative values. Fig. 9 shows the BIC scores for the two datasets across varying numbers of GMM components and covariance types. The covariance matrix types considered in decreasing order of complexity were: full, diagonal, spherical (isotropic); as well as tied (where all components share the same covariance matrix). We tested models with one to ten components. For the Landsat dataset, the optimal BIC was achieved with two components, which we selected for clustering. For the Default dataset, the optimal BIC was found at ten components; however, to prioritize simplicity, we chose seven components, as this yielded a BIC only slightly higher than the minimum.

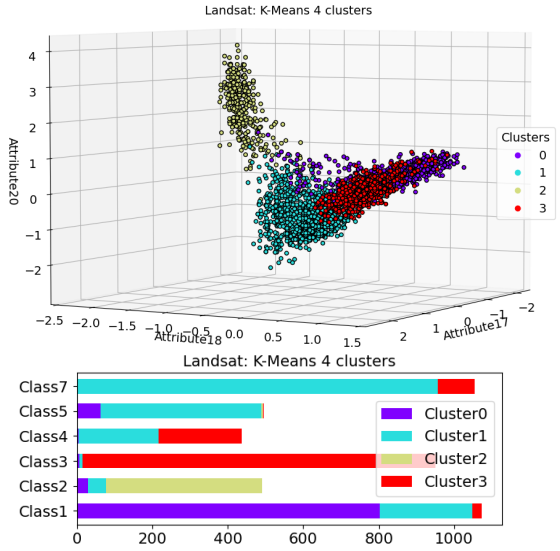

Fig. 7: K-means clustering of the Landsat dataset with 4 clusters: data scatter plot and bar plot of class proportions per cluster. Note: the bar plot horizontal axis is the number of samples.

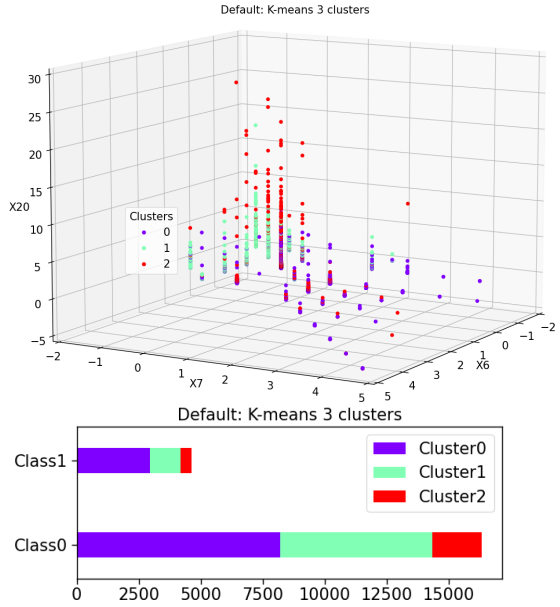

Fig. 8: K-means clustering of the Default dataset with 3 clusters: data scatter plot and bar plot of class proportions per cluster. Note: the bar plot horizontal axis is the number of samples.

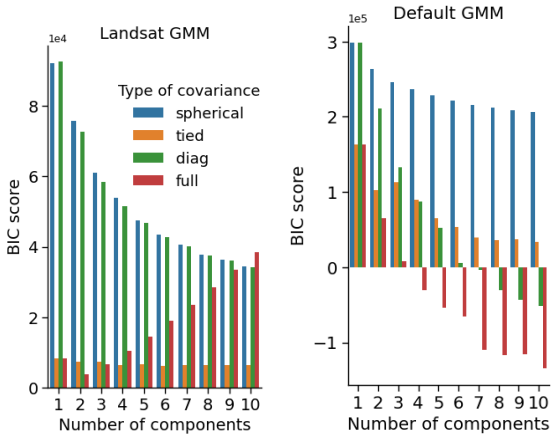

Fig. 9: Gaussian mixture model tuning for the Landsat and Default datasets.

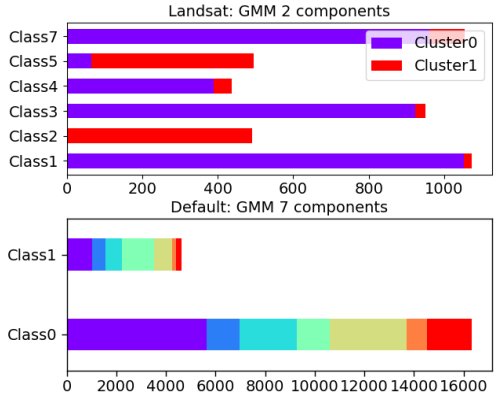

Fig. 11: Class assignment to the clusters obtain from the Gaussian mixture model of the Landsat (2 components) and the Default (7 components) datasets. Note: the bar plot horizontal axis is the number of samples.

Fig. 10 presents the scatter plot with cluster labels generated for the Landsat dataset, while Fig. 11 shows the ground truth class assignment to the clusters generated for the two datasets. The clustering results for the Landsat dataset are reasonable with the two left-most classes in the ground truth scatter plot (see Fig. 1) being grouped into one cluster, and the four right-most classes forming the second cluster. In contrast, the Default dataset's clustering results are more difficult to interpret due to the larger number of clusters. Notably, there does not appear to be a strong preference for any cluster to align with a specific class. the behavior is the same if instead of seven we use two components in the GMM.

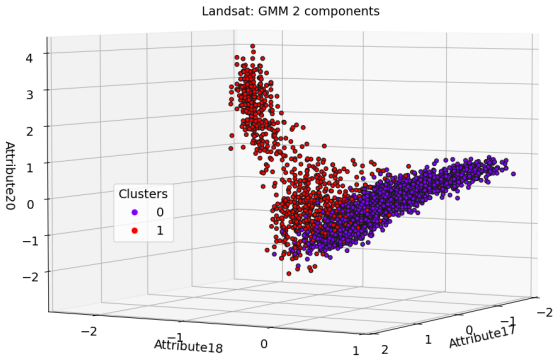

Fig. 10: Scatter plot of the Landsat dataset clustered with EM using 2 Gaussian mixture components.

Both k-means and EM performed reasonably well on the Landsat dataset, while both showed similarly poor performance on the Default dataset. The fewer clusters produced by GMM for the Landsat dataset compared to k-means can be attributed to the use of the Bayesian Information Criterion (BIC) in GMM. BIC penalizes model complexity, favoring fewer components unless additional components significantly improve model fit. Conversely, the higher number of clusters generated by GMM for the Default dataset likely reflects the dataset's greater variability and less-defined structure, requiring more mixture components to model the data and improve the log-likelihood.

IV. Clustering after Dimensionality Reduction

In this section, we first explore dimensionality reduction on the two datasets using: 1) PCA, 2) ICA, and 3) RP. We then apply the clustering algorithms to the projected data and compare the results with those obtained using the full-dimensional data.

A. Dimensionality Reduction

As part of our data exploration, we calculated the correlation coefficients between the features of both datasets. We observed a number of highly correlated features in both datasets, with the Landsat dataset having more correlated features than the Default dataset. Based on this, we hypothesize that the resulting PCA directions will be fewer for the Landsat dataset. However, since uncorrelated features are not necessarily statistically independent, it is more challenging to predict the number of ICA components. For RP, we anticipate that with a sufficient number of repetitions, the randomized projections can potential identify a similar number of significant components as in PCA or ICA. Nonetheless, RP is expected to stand out for its computational efficiency.

1) PCA

In PCA, the explained variance ratio of a component is computed as the ratio of its eigenvalue to the sum of all eigenvalues. To determine the number of principal components for dimensionality reduction, we plotted the cumulative explained variance ratio against the number of components (see Fig. 12) and selected the smallest number of components that can collectively explain 95% of the total variance. This resulted in 6 principal components for the Landsat dataset and 16 for the Default dataset. These values represent the effective rank of the feature matrix based on the 95% threshold, indicating

that the Landsat dataset has 6 uncorrelated components out of 36 features, while the Default dataset has 16 uncorrelated components out of 25 features leading to greater dimensionality reduction for the case of the Landsat dataset.

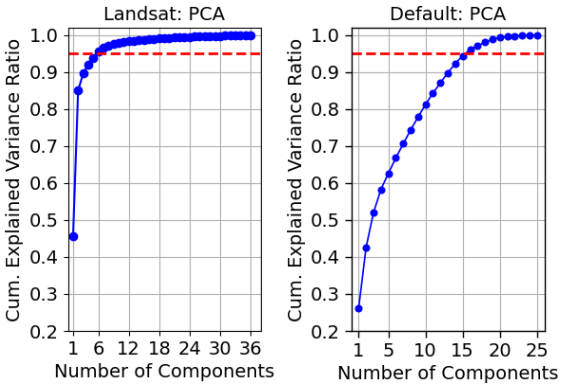

Fig. 12: Cumulative explained variance ratio of PCA components in the Landsat and Default datasets.

2) ICA

ICA aims to separate data into statistically independent components, which are typically non-Gaussian. In the scikit-learn implementation used, ICA is preceded by PCA to reduce the dimensionality to the desired number of components before applying the ICA algorithm [3]. To assess the non-Gaussianity of the resulting components, we calculated the average absolute kurtosis across the components and identified the number of components that maximized this value. A higher average kurtosis indicates stronger non-Gaussianity in components collectively, guiding the selection of the optimal dimensionality. Fig. 13 shows the average kurtosis as a function of the number of independent components for the two datasets. Based on the results, we selected 11 components for the Landsat dataset and 4 components for the Default dataset.

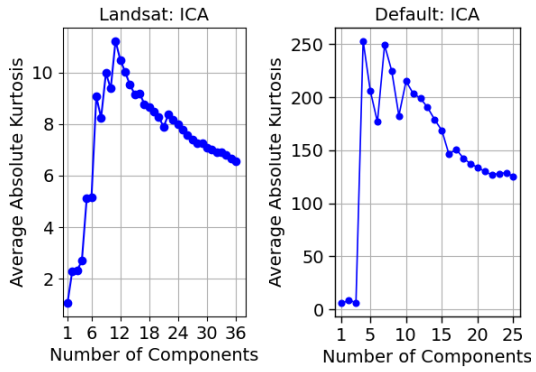

Fig. 13: Average absolute kurtosis of the ICA components in the Landsat and Default datasets.

3) RP

The RP method was applied using the sparse random matrix approach. To select the optimal number of components, we calculated the Frobenius norm of the reconstruction error (the difference between the original and reconstructed feature matrices), normalized by the norm of the original feature matrix. Due to the randomization involved, we ran the RP procedure 100 times with different random seeds. For each number of components, we selected the result from the random seed that minimized the reconstruction error. Fig. 14 illustrates the minimum reconstruction error for each number of components and illustrates the range between the minimum and maximum reconstruction error across the multiple random seeds. As shown in the figure, there is no elbow point that indicates an optimal number of components but the error decreases almost parabolically with the addition of components. For further using RP for clustering, we accepted a reconstruction error of 20% resulting to 31 components for Landsat and 23 components for the Default datasets reducing the number of dimensions but just 5 and 2 respectively. The same result was obtained using the Gaussian RP approach instead of the sparse.

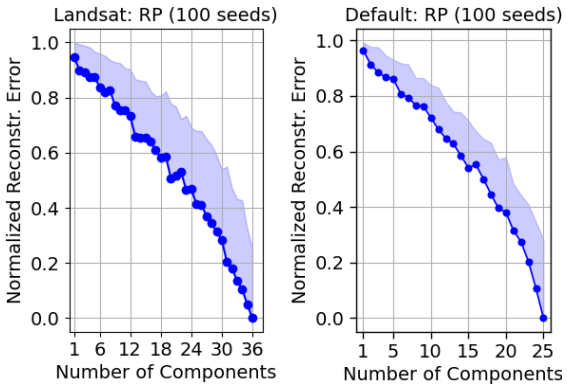

Fig. 14: Normalized reconstruction error of the ICA components in the Landsat and Default datasets.

Table I compares the normalized reconstruction error and the wall-clock training time for each method. Although ICA is not designed to minimize reconstruction error, evaluating it provides a consistent way to assess how well the reduced-dimensional representations preserve the original data. For the Landsat dataset, all three methods achieve a reconstruction error below 0.2. However, for the Default dataset, ICA with 4 components exhibits a significantly higher error. This is acceptable since ICA prioritizes maximizing statistical independence over reconstruction accuracy. Additionally, RP stands out as the fastest method, requiring significantly less training time than PCA and ICA. The training time relationship between PCA and ICA varies depending on the number of components. "Our initial hypothesis that the Landsat dataset would require fewer PCA components than the Default dataset was confirmed. Similarly, our expectation about the efficiency of the RP method was validated. However, the unexpectedly large number of components required by RP was unforeseen. This suggests that with more random seeds, better projections might be discovered, potentially reducing the number of necessary components.

TABLE I: Comparison of dimensionality reduction methods

Landsat dataset (Train data: 4,504×36)			Default dataset (Train data: 20,975×25)		
Method	Norm. Recon. Error	Train Time	Method	Norm. Recon. Error	Train Time
PCA 6 comps	0.210	0.008	PCA 16 comps	0.199	0.092
ICA 11 comps	0.137	0.046	ICA 4 comps	0.646	0.052
RP 31 comps	0.204	0.002	RP 23 comps	0.203	0.003

B. Unsupervised Clustering using Reduced Dimensions

In this section, we assess the performance of dimensionality reduction algorithms by evaluating how well their projected data can be clustered using K-means and GMM. The number of components determined in the previous section was applied. For each projection, the number of clusters for both K-means and GMM was tuned independently in an unsupervised manner using the procedure outlined in Section III (omitted here for brevity). Note that for all GMM runs, the full covariance matrix type consistently yielded the best BIC score.

Table II provides a summary of the number of clusters identified for each combination of projection and clustering algorithm across the two datasets. It also presents the V-measure scores, which evaluate the clustering performance relative to the ground truth, as well as the wall clock time required to train the clustering algorithms in each case. The following observations can be drawn.

In the Landsat dataset, all clustering results were found be reasonable when considering the scatter plots, cluster assignment bar plots, and V-measure scores. As shown in Table IIa, the best V-measure score was achieved using K-means on the data projected by ICA. The corrsponding cluster assignment bar plot is shown in Fig. 15. Compared to the clustering results in Fig. 7 (K-means on the original data), the main improvement with ICA is that Class 1 was more effectively grouped into a single cluster containing only data points from Class 1. Another observation is that When K-means was applied to data projected by PCA, the clustering result was virtually identical to that in Fig. 7. This shows that PCA was effective in considerably reducing the number of dimensions without hurting the clustering perfomance of K-means. An additional observation is that GMM produced a more realistic clustering solution when applied to the data projected by PCA and ICA, identifying four clusters instead of the two clusters that GMM found in the original data. This improvement is likely due to the reduced dimensionality, which simplified the covariance matrix parameters in GMM, leading to better clustering performance. Lastly, the RP that resulted to the smallest dimensionality reduction yielded GMM clusters identical to those of the original dataset.

In the Default dataset, none of the linear projections with any clustering method were able to identify the ground truth classes. This was not surprising, as the ground truth class labels are not inherently structured or well-separated. Fig. 16 plots the PCA projected data against the three first principal axes of PCA. It can be seen that While PCA results in visually distinct clusters, the clustering does not align with the ground truth classes. Another interesting observation is that GMM consistently identified more clusters than K-means in all cases. This suggests that GMM struggled to find a low-cluster solution that minimizes the BIC score effectively. Additionally, GMM is more sensitive to outliers, which may lead it to create more clusters. In contrast, K-means, with its hard boundary

TABLE II: Impact of dimensionality reduction on the performance of the clustering algorithms for the two datasets.

Landsat dataset				
Projection	Clustering	Clusters	V-Meas.	Train Time
Original 36 feat.	K-means	4	0.56	0.33
	GMM	2	0.32	2.53
PCA 6 comps	K-means	4	0.56	0.33
	GMM	4	0.57	1.54
ICA 11 comps	K-means	4	0.63	0.35
	GMM	4	0.55	2.42
RP 31 comps	K-means	4	0.43	0.33
	GMM	2	0.32	2.46
Default dataset				
Projection	Clustering	Clusters	V-Meas.	Train Time
Original 25 feat.	K-means	3	0.01	0.29
	GMM	7	0.03	51.60
PCA 16 comps	K-means	3	0.01	0.73
	GMM	9	0.01	45.23
ICA 4 comps	K-means	5	0.05	0.31
	GMM	6	0.02	8.14
RP 23 comps	K-means	3	0.02	0.28
	GMM	8	0.03	54.80

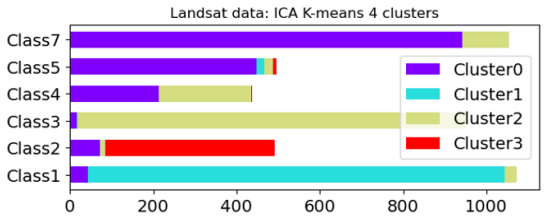

Fig. 15: Landsat clustering with k-means and 4 clusters after ICA projection with 11 components. Note: the bar plot horizontal axis is the number of samples.

approach, is often better at ignoring extreme outliers.

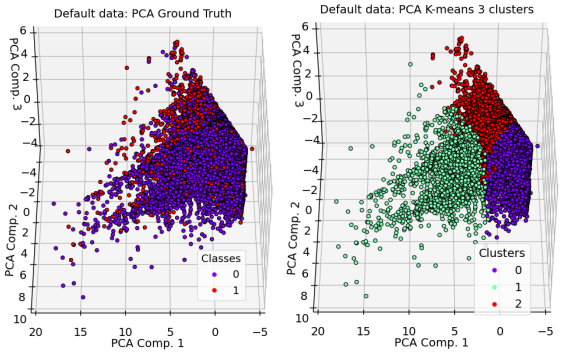

Fig. 16: Default data after PCA projection; ground truth labels (left) and labels from k-means clustering (right).

Regarding wall clock times needed for clustering, GMM was significantly more time-consuming than K-means, even when both algorithms used the same number of clusters. Additionally, the time complexity of GMM was more sensitive to the number of projection dimensions than K-means. As the number of projection dimensions decreased, the computational cost of GMM also decreased. For example, as shown in Table II, clustering data projected to 11 dimensions (using ICA) took 2.42 seconds, while clustering data projected to 6 dimensions (using PCA) took only 1.54 seconds. In contrast, K-means runtime remained relatively constant across all cases. This difference likely arise because GMM requires the evaluation and inversion of full covariance matrices, which is more computationally intensive than the distance calculations used in K-means.

V. Neural Network Learner

In this section, we investigate how the performance of a neural network (NN) learner is affected by: 1) the application of dimensionality reduction methods, and 2) the introduction of new features generated by clustering algorithms. This study focuses solely on the Landsat dataset, as the Default dataset did not yield meaningful results with the clustering methods examined. Since the NN trained on the original dataset achieves a test ROC-AUC score of around 0.98, we do not expect our experiments to have significant changes in the test score. However, we hypothesize that it may be possible to achieve similar performance using a simpler network architecture, and potentially with fewer learning iterations when the unsupervised learning techniques are used. For reference, the NN model tuned on the original dataset consists of a single hidden layer with 12 nodes and uses the ReLU activation function.

A. Neural Network leveraging Dimensionality Reduction

The NN model was independently tuned for the training data projected by each of the PCA, ICA, and RP methods, using the number of components specified in the previous section (see Table I). Five-fold cross-validation was employed, and the hyperparameters explored included the number of hidden layers, the number of nodes per hidden layer, and the activation function. The analysis indicated that, for all cases, a single hidden layer was sufficient, and the ReLU activation function consistently produced the best validation scores. Fig. 17a presents the mean ROC-AUC cross-validation scores as a function of the number of hidden nodes. Interestingly, contrary to our hypothesis, dimensionality reduction did not simplify the model architecture. This can be attributed to the fact that all three projection methods yielded normalized reconstruction errors of approximately 0.20, retaining most of the original data's complexity. For the final model, we selected 12 hidden nodes. The test set was transformed in each case using the same projection matrix derived from the corresponding dimensionality reduction method applied to the training data.

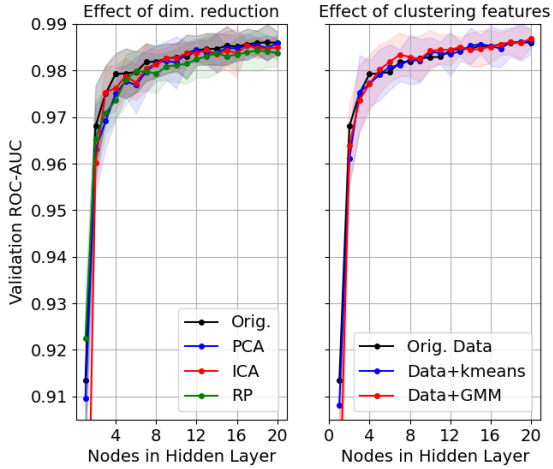

Fig. 17: Validation scores of the Landsat neural network as a function of the number of hidden nodes: (a) impact of different dimensionality reduction methods; (b) impact of additional features from clustering algorithms. Notes: The score for the ICA model with one hidden node was omitted to enhance the clarity of the figure. The plots share the same y-axis.

Table III summarizes the training wall clock time, the number of training iterations to convergence, and the training and test scores for each projected dataset. We observe that PCA, ICA, and RP slightly decreased the training score while improving the test score compared to the original data. This likely occurs because reducing the dimensionality removes some noise from the data, helping the model avoid overfitting to the training set. The improvement in the test score is slightly more pronounced for PCA and ICA, which result in significantly fewer dimensions than RP.

Another key observation is that the neural network required fewer iterations to converge when using the projected data, leading to shorter training wall clock times. This supports our initial hypothesis. Fig. 18a illustrates the iterative learning curves of the neural network for the four cases, showing that ICA achieved the fewest convergence iterations and thereby the highest time efficiency. The reduced dimensionality has evidently enabled gradient descent to navigate more quickly

toward the minimum loss, and the independent directions identified by ICA likely further optimized the solution space for the learning algorithm.

TABLE III: Comparison of dimensionality reduction methods

Data	Train Time	Converg. Iteration	Train roc-auc	Test roc-auc
Orig. (36)	8.298	906	0.992	0.981
PCA (6)	6.941	838	0.989	0.983
ICA (11)	5.802	655	0.989	0.983
RP (31)	6.769	712	0.990	0.982
Orig.+ k-means (40)	7.212	793	0.993	0.982
Orig. + GMM (38)	7.263	804	0.993	0.980

Note: the numbers in parentheses are the data dimensions.

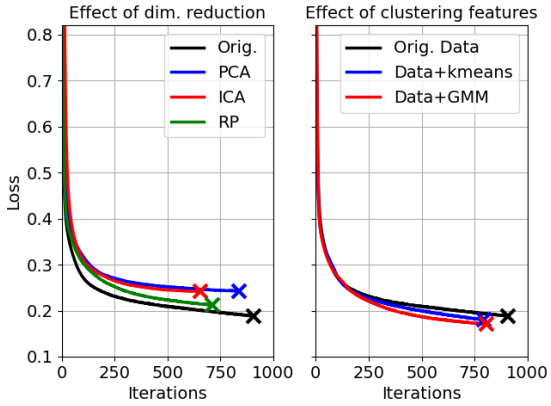

Fig. 18: Iterative learning curves of the Landsat neural network model: (a) impact of different dimensionality reduction methods; (b) impact of additional features from clustering algorithms. Note: the plots share the same y-axis.

B. Neural Network leveraging Unsupervised Clustering

We generated new features from the unsupervised clustering results and added them as additional features to the original (unprojected) features of the Landsat dataset. Two scenarios were evaluated: one utilizing clustering results from k-means and the other using results from GMM. The models developed in Section III were used, namely the k-means model with four clusters (see Fig. 7) and the GMM model with two clusters (see Fig. 7). For the k-means features, we used the cluster labels as categorical features, which were one-hot encoded to produce four additional features. For the GMM features, we used the probability density values of each mixture component for each data point, adding two continuous features. For the test set, the new features were generated by applying the trained k-means and GMM models to the test data.

As shown in Fig. 17b, adding the new features had minimal impact on the cross-validation score of the original model, leaving it nearly unchanged compared to the original dataset. However, a slight improvement was observed in the range of 5 to 7 hidden nodes when GMM features were included—an effect not seen in the dimensionality reduction scenario. This suggests that our hypothesis regarding the potential for a simpler model is partially validated. We believe this outcome would be more pronounced if the clustering algorithm had accurately captured the ground truth clusters.

For the final training, we again used 12 hidden nodes and the ReLU activation function. As shown in Fig. 18b and Table III, the NN converged in fewer iterations and with shorter wall clock times after incorporating the clustering features. In terms of performance, both models achieved the same training score; however, the k-means model produced a slightly higher test score. This may be attributed to k-means identifying a greater number of clusters, capturing more information relevant to the ground truth solution.

VI. Conclusions

In the Landsat dataset, both the k-means and GMM clustering algorithms were able to produce reasonable clustering results, as initially hypothesized. The GMM was more conservative than k-means, generating fewer clusters, likely due to its preference for simplicity, guided by the minimization of the BIC score. The PCA and ICA projections helped the GMM in capturing the same clustering patterns identified by k-means, while ICA further enhanced k-means' performance, compared to the unprojected data. The RP method did not help in the clustering as it was unable to reduce the problem's dimensions as effectively as the other methods.

The Default dataset lacked well-defined ground truth clusters, which led to poor performance by the clustering algorithms. As it was expected, the linear projections were unable to map the data into a space where the classes were effectively separated, making them unsuitable for clustering this dataset. To address this, it would be worthwhile to explore nonlinear dimensionality reduction techniques, such as the UMAP algorithm, which may better capture the underlying structure of the data.

When training the NN model on the Landsat dataset, applying dimensionality reduction reduced both the number of iterations required for convergence and the training wall clock time, which confirmed our initial hypothesis. A similar effect was observed when adding new features extracted from clustering. However, our hypothesis that dimensionality reduction would simplify the NN model was not supported by the results, suggesting that dimensionality reduction preserved the complexity of the problem. Nonetheless, adding features from the GMM clustering algorithm provided some limited evidence that the NN model could be simplified by leveraging unsupervised clustering data, as initially hypothesized. To explore this possibility further, a more effective clustering model and a dataset that does not produce such accurate NN

results would be needed, in order to create a scenario with sufficient room for improvement.

VII. References

[1] Yeh, I. (2009). Default of Credit Card Clients [Dataset]. UCI Machine Learning Repository. https://doi.org/10.24432/C55S3H.

[2] Srinivasan, A. (1993). Statlog (Landsat Satellite) [Dataset]. UCI Machine Learning Repository. https://doi.org/10.24432/C55887.

[3] Scikit-learn: Machine Learning in Python, Pedregosa et al., JMLR 12, pp. 2825-2830, 2011.