CS 7641 Machine Learning:
Unsupervised Learning and Dimensionality Reduction

Jilong Cui

jcui@gatech.edu

1 OVERVIEW

This report presents an analysis of two clustering algorithms, KMeans and Expectation Maximization (EM), combined with four dimensionality reduction techniques: Principal Component Analysis (PCA), Independent Component Analysis (ICA), Randomized Projections, and Lasso Feature Selection. These methodologies are systematically applied and fine-tuned on two distinct classification datasets, each with its own unique characteristics. Subsequently, the datasets processed through these dimensionality reduction and clustering pipelines are employed to train Neural Network classifiers. The performance outcomes, as well as the behavior and implications of each method on the datasets, are examined and detailed in the analysis.

2 DATA SET

The datasets chosen for this investigation – UFC Fight and Wine – present distinct challenges.

The **UFC Fight Dataset** provides the nuances of mixed martial arts, capturing 156 features that represent a fighter's performance and characteristics in a binary classification problem—predicting the winner from either the red or blue corner. The dataset, comprising 3,592 entries, reflects the complexities of competitive sports, where outcomes are not solely determined by quantitative measures but are also influenced by the unpredictable nature of human competition. The data reveals a pronounced imbalance, with the red corner—typically the favorite—emerging as the victor in 66.26% of the cases. This imbalance underscores the significance of accurately predicting blue corner victories.

The dataset's attributes, a mixture of continuous and binary data, reflect the diverse elements of a fighter's skill set and strategic approach, indicating that clustering could provide valuable insights into different fighting styles and techniques, beyond the binary outcome of wins and losses. Using f1_score (macro) as a metric allows for a more balanced consideration of predictive performance across both classes, particularly valuing the less represented blue corner.

On the other hand, the **Wine Dataset**, derived from a chemical analysis of wines from three cultivars in Italy, offers a completely different landscape. With 13 continuous features and 5,000 entries (selected for computational efficiency), the dataset invites a three-class classification to discern the cultivar origin of each wine. The features here are objective, quantifiable, and predictable, making for a less complex classification task than the UFC data, as evidenced by higher achievable accuracies (over 90%). The balanced nature of the dataset, in conjunction with the use of f1_score (weighted) as an optimization metric, provides a fair representation of each class in the predictive modeling process.

Clustering within the Wine Dataset could be particularly effective in identifying inherent groupings based on chemical compositions, potentially revealing subtler distinctions between the cultivars that might not be immediately apparent. This approach contrasts with the UFC dataset, where clustering could unravel the layered strategies and combat styles of fighters, which are not directly linked to the fight's outcome.

In summary, while the UFC Fight Dataset offers a fertile ground for clustering analyses to understand fighter profiles and tactics, the Wine Dataset is well-suited for clustering based on chemical properties.

3 CLUSTERING ON ORIGINAL DATA SET

KMeans and Expectation Maximization (EM) are implemented by python sklearn KMeans and GaussianMixture.

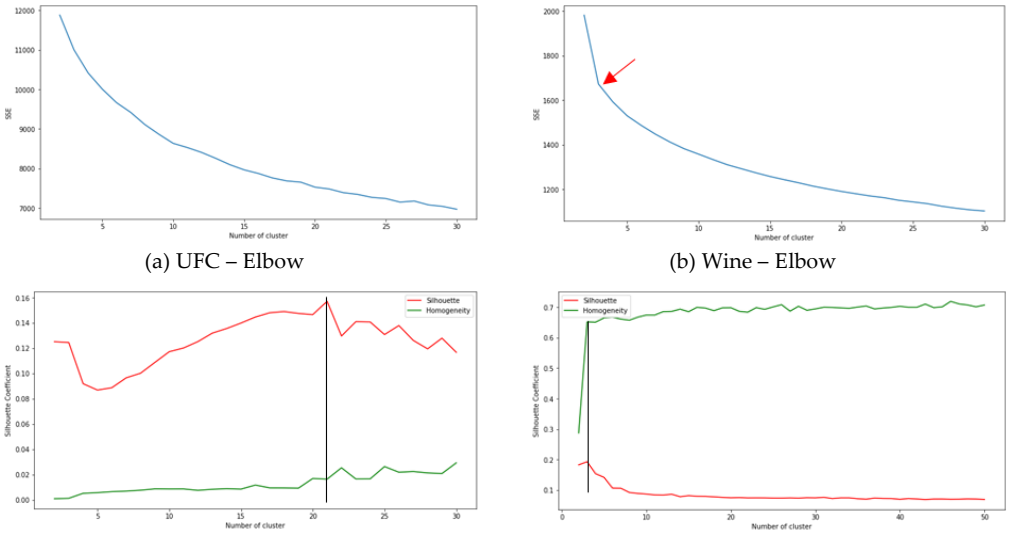
For determining the number of clusters in both methods, the elbow method is a common technique that involves plotting the explained variation as a function of the number of clusters and picking the elbow of the curve as the number of clusters to use. Figure 1(a) and (b) presents the elbow graph for UFC and Wine dataset. In the UFC dataset, the drop in error exhibited an inversed log scale, suggesting a gradual decline in benefit from increasing the number of clusters. The Wine data showed a more definitive elbow at 3 clusters, which aligns with the number of cultivars, and then continued to drop in an inversed log scale.

However, to more accurately capture the effectiveness of the clustering, the Silhouette coefficient and homogeneity score were used. The Silhouette coefficient measures how similar an object is to its own cluster compared to other clusters. The homogeneity score indicates if all of its clusters contain only data points which are members of a single class. These metrics were plotted against the number of clusters, revealing optimal clustering where the Silhouette coefficient is maximized and the homogeneity score is reasonably high. For the UFC data, the optimal number of clusters was found to be 22 (Figure 1 (c) and (e)), whereas for the Wine data it was 3 (Figure 1 (d) and (f)), which corresponds with the actual number of cultivars.

When clusters were set to the number of classes in the original labels (2 for UFC, 3 for Wine), the match scores—indicating how well the clusters corresponded to the original labels—were computed for both KMeans and EM:

- KMeans UFC match score was relatively low at 0.429, possibly due to the complexity and high-dimensionality of the data which KMeans struggles with, especially when there are imbalances in cluster sizes and non-spherical shapes.
- KMeans Wine match score was high at 0.903, reflecting that KMeans can perform well on data with clear, well-separated clusters.
- EM UFC match score was higher at 0.663, which might be due to its ability to incorporate the probability of cluster memberships and handle the imbalances and complexity in the data more effectively than KMeans.
- EM Wine match score was slightly lower at 0.87, which could be due to the method's tendency to sometimes converge to local optima or due to its probabilistic nature which may not be as effective as KMeans in this particular scenario where the data has well-defined clusters.

In summary, the performance of KMeans and EM can vary significantly based on the dataset's characteristics, with EM generally offering a more flexible approach that can lead to better clustering on complex datasets like UFC, while KMeans may be preferable for more clearly defined and separated data, as seen with the Wine dataset.



(a) UFC – Elbow (b) Wine – Elbow

(c) UFC –KMean Coef. (d) Wine – KMean Coef.

Figure 1. Clustering Analysis – KMean & Expectation Maximization (EM)

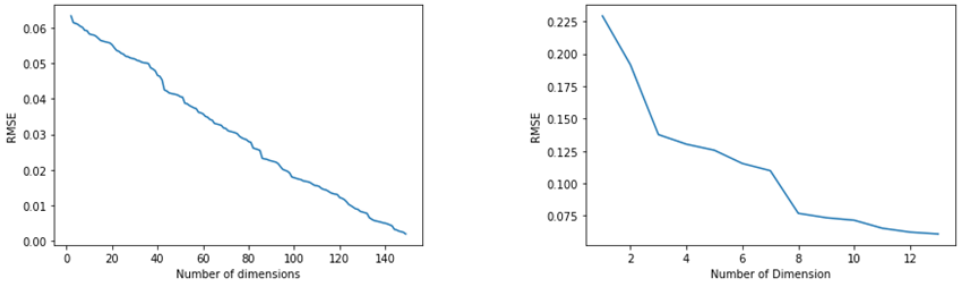
4 DIMENSIONALITY REDUCTION ON DATA SET

In the dimensionality reduction phase of the analysis, four algorithms were employed: Principal Component Analysis (PCA), Independent Component Analysis (ICA), Gaussian Random Projection (GRP), and Lasso feature selection.

The reconstruction error was a key metric used to evaluate the quality of dimensionality reduction, which measures the loss of information when projecting the data to a lower-dimensional space and then reconstructing it back to the original space. The reconstruction error analysis was plotted against the number of dimensions retained. Figure 2 shows the graph for GRP as an example.

The decision on the number of dimensions to reduce to was made by combining the results of the reconstruction error analysis with the intrinsic properties of the datasets and domain knowledge. For instance, domain knowledge about the significance of certain chemical properties in wine could inform the selection of dimensions in the Wine dataset.

Table 1 summarizes the outcomes of the dimensionality reduction process, showing the number of dimensions each method reduced the datasets to and the associated reconstruction errors.



(a) UFC – Randomized Projections (d) Wine – Randomized Projections

Figure 2. Reconstruction Error vs. Dimensionality (Randomized Projections as Example)

Table 1. Summary of Dimensionality Reduction

		Original	PCA	ICA	RP	Lasso
UFC	**Features**	156	72	80	40	39
	Reconstruction Error	-	0.04	0.027	0.042	-
Wine	**Features**	13	8	3	9	4
	Reconstruction Error	-	0.19	0.163	0.075	-

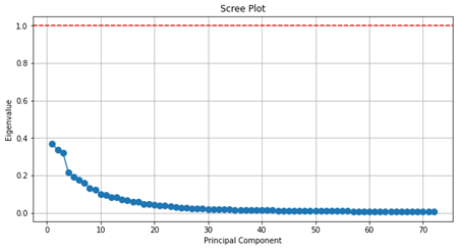
4.1 Principal Component Analysis (PCA)

PCA reduces dimensionality by transforming the original features into a smaller number of uncorrelated variables called principal components.

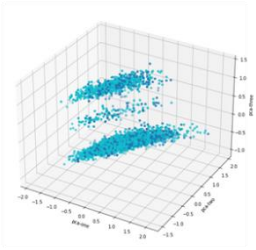
For the UFC dataset, PCA was applied to reduce the feature space from 156 to 72 while maintaining a reconstruction error of 0.04. The rationale behind the low reconstruction error for the UFC dataset could be due to the intrinsic structure of the data where certain combinations of the features capture the main variance and hence the essence of the data. This might suggest that while the dataset has 156 features, the actual informative dimensionality is much lower.

Figure 3(a) shows the eigenvalue distribution. Typically, eigenvalues in PCA provide a measure of the variance explained by each principal component. The distribution for the UFC dataset showed that the first three components were significantly more important than the rest, with a noticeable drop in eigenvalue size around the 20th component, before approaching zero. This eigenvalue behavior aligns with the cluster analysis that suggests 20-25 clusters could capture the underlying structure well.

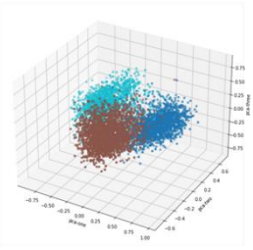
In contrast, the Wine dataset, originally with 13 features, was reduced to 8 with a higher reconstruction error of 0.19. This error is acceptable, considering that less information is lost and the core characteristics that distinguish the three classes of wine are retained. The PCA eigenvalue distribution for the Wine dataset showed dominance in the first three components, with subsequent values nearing zero, indicating that most of the dataset's variability can be encapsulated in just a few components.



(a) UFC – PCA eigenvalue



(b) UFC – PCA first three PC



(a) Wine – PCA first three PC

Figure 3. PCA Eigenvalue and Principal Component Distribution

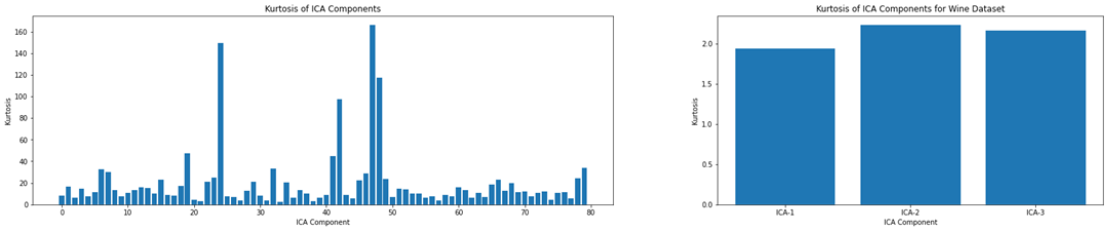
3D plots of the datasets using the first three principal components (Figure 3 b for UFC and Figure 3 c for Wine) provide a visual representation of the data in the reduced space. For the UFC dataset, the plot revealed that there are broadly three groups; however, these do not correspond to the winners and losers (red and blue corners). This suggests that PCA on UFC data is more effective at revealing the fighters' styles or strategies rather than predicting match outcomes. On the other hand, the Wine dataset displayed a clear separation between the three classes in the 3D plot, indicating that PCA has successfully captured the underlying structure of the data, which corresponds well to the original classification labels.

4.2 Independent Component Analysis (ICA)

ICA separates a multivariate signal into additive, independent non-Gaussian signals.

For the UFC dataset, ICA managed to reduce the original 156 features to 80 with a reconstruction error of 0.027. The lower error compared to PCA might be due to ICA's ability to find a basis along which the data points are statistically independent, which can be more informative in certain cases compared to the uncorrelated basis that PCA finds. The Wine dataset saw a more dramatic reduction in dimensionality with ICA, going from 13 features to only 3, and yielding a reconstruction error of 0.163. This suggests that the few independent components found by ICA encapsulate the majority of the information needed to differentiate between the classes of wine. The relatively low reconstruction error with just three features could be attributed to the fact that the original features of the Wine dataset may have a certain degree of redundancy, and the non-Gaussian nature of ICA helps to capture the essence of the data without much loss of information.

Kurtosis is a statistical measure used to describe the distribution of observed data points. Figure 4 (a) demonstrating kurtosis distribution showed that for the UFC dataset, similar to PCA, there are a few significant independent components with the rest contributing less, hence the decision to cut down to 80 features. However, the principle component plot indicated that the labels are mixed, suggesting that while ICA can separate data into stylistically different groups, it is not effective in distinguishing winners. For the Wine dataset, the kurtosis values (Figure 4 (b)) of the three features being very close to each other imply that they contribute almost equally to the separation of the data points. The principle component plot likely showed a clear distinction among the three classes, mirroring the effective class separation observed with PCA. This clear delineation among the classes confirms that ICA can successfully reduce dimensionality while still allowing for accurate classification in the Wine dataset.



(a) UFC – ICA kurtosis (b) Wine – ICA Kurtosis

Figure 4. ICA Kurtosis and Independent Component Distribution

4.3 Gaussian Random Projection (GRP)

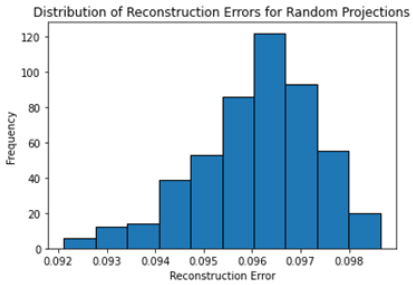
Gaussian Random Projection reduces dimensionality through a probabilistic process.

For the UFC dataset, GRP was able to reduce the feature space to 40 from 156 with a reconstruction error of 0.042, while the Wine dataset was reduced to 9 from 13 features with an error of 0.075. This technique is particularly effective when the data lies in a high-dimensional space because it relies on the random nature of the projections, which can, on average, capture the structure of the data without having to compute the pairwise distances explicitly. It's effective because it does not require the preservation of all pairwise distances but rather the preservation of the overall structure of the data.

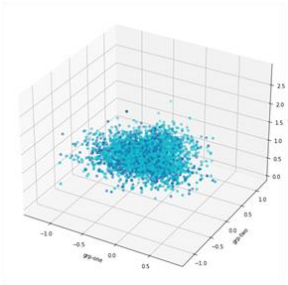
Figure 5 (a), which shows the error distribution over 500 trial runs with consistent projection, indicates a Gaussian distribution because the random projection inherently involves a Gaussian matrix. The Central Limit Theorem may contribute to this result, as the sum of many random projections (each entry of the matrix being a random variable) would tend to a Gaussian distribution.

The visualization of the data with the first three principal components would still show the UFC data as one trunk with labels mixed together due to the inability of GRP to capture the complex patterns needed to separate the winners. This indicates that while dimensionality can be reduced, the remaining features are not discriminative enough for classification tasks like predicting UFC winners.

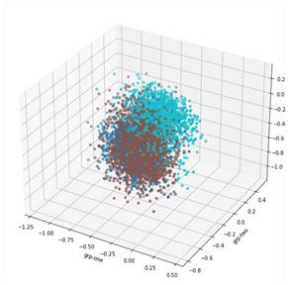
In contrast, the Wine dataset appears to be better separated into three distinct groups, even after the reduction. This may be because the inherent differences between the wine classes are more pronounced and can be preserved even after projecting the data into a lower-dimensional space. Therefore, the Random Projection method can still maintain the separation between the classes, reinforcing the notion that the Wine dataset has an intrinsic division that aligns well with its three classes. This separation lends itself well to clustering and classification, as seen in the distinct grouping observed in the visualization.



(a) UFC – RP error distribution



(a) UFC – RP first three PC



(a) Wine – RP first three PC

Figure 5. Randomized Projection Reconstruction Error and Principal Component Distribution

4.4 Lasso feature selection

Lasso selects features by penalizing the absolute size of the coefficients in regression analysis. One of the key properties of Lasso is its ability to perform feature selection by driving some coefficients to zero, effectively eliminating them from the model.

In the case of your datasets: for the UFC dataset, Lasso selected 39 features, for the Wine dataset, Lasso selected 4 features.

The nature of Lasso means that it doesn't reconstruct the original data; rather, it identifies a subset of features that contributes most significantly to the output variable. The "error" here is not about reconstruction (like in PCA or ICA) but about prediction. Applying Lasso for feature selection is trying to minimize the prediction error while also penalizing the model complexity. This predictive error minimization does not necessarily capture the full structure of the data but rather focuses on the aspects of the data that are most predictive of the target

variable. This is why Lasso can effectively reduce the number of features while maintaining or sometimes even improving the predictive performance of the model.

The selected features from the Lasso model will be examined in the clustering section to assess their impact on clustering performance. The assumption is that the features retained by Lasso should carry significant information for clustering, as they have been identified as influential in predicting the outcome. However, the correlation with the outcome doesn't always mean a good separation in the feature space, which is essential for clustering. The results from clustering with these Lasso-selected features will reveal how well this assumption holds.

5 CLUSTERING ON REDUCED DIMENSION DATA SET

Table 2 summarizes the clustering analysis results for the 16 cases with the matching score to the original label. PCA and Lasso appear particularly effective for the UFC dataset when used in conjunction with KMeans, substantially enhancing the matching scores by about 30%. This improvement suggests that these techniques are successful in condensing the feature set to the most relevant variables for cluster differentiation. For the wine dataset, PCA and ICA similarly yield slight improvements in matching scores, indicating these methods are capable of isolating key features that define cluster boundaries. The stability of EM's scores across different models can be attributed to its probabilistic nature. EM considers a mixture of distributions to model the data, and thus may be more robust to changes in feature space since it accommodates the probability of a data point belonging to each cluster instead of assigning a hard classification.

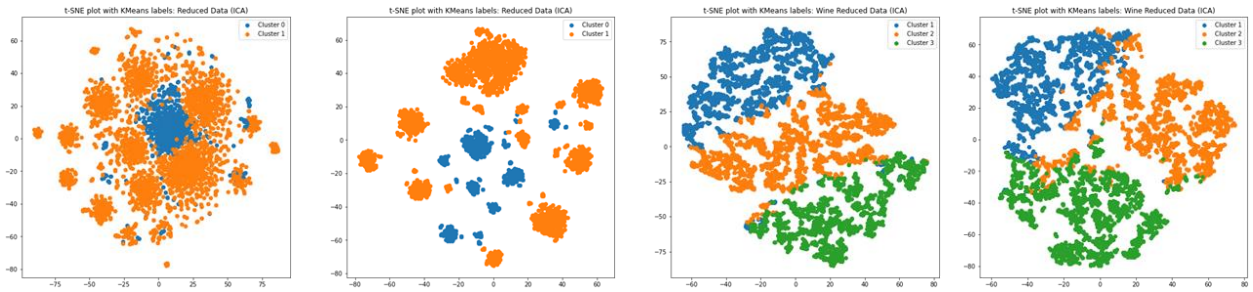
t-Distributed Stochastic Neighbor Embedding (t-SNE) is a technique that aims to reduce the dimensionality of data for visualization purposes, while preserving the local structure of the data as much as possible. It tends to expand clusters that are tight in high-dimensional space and contract clusters that are sparse, which can make the overall data structure easier to interpret visually.

For UFC (Figure 6 a b c), ICA for the UFC dataset seems to homogenize the data under a single label, which may indicate that the components identified by ICA are not capturing the underlying cluster structure well. This could be due to the UFC dataset having clusters that are not well-separated by statistically independent features. Lasso, with its emphasis on sparsity and selecting the most predictive features, appears to maintain the inherent clustering structure in the UFC dataset while enhancing cluster separation.

For Wine data set (Figure 6 d e f), ICA appears to achieve a clear demarcation of the three expected groups, perhaps because the underlying factors that differentiate the wine varieties are well-modeled by independent components that ICA extracts. Lasso's tendency to keep the original data distribution yet improve the cluster formation suggests that the few features it selects are significant for the underlying class distinctions.

Table 2. Summary of Clustering on Reduced Dimension Data Set (Matching Score with Original Label)

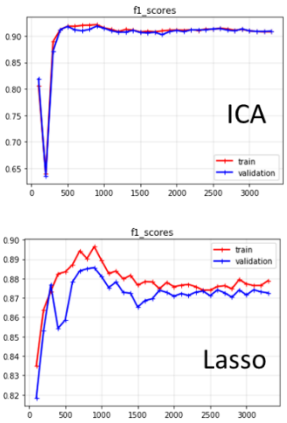
		Original	PCA	ICA	RP	Lasso
UFC	**Kmean**	0.429	0.571	0.604	0.403	0.597
	EM	0.663	0.663	0.663	0.663	0.663
Wine	**Kmean**	0.903	0.902	0.911	0.799	0.813
	EM	0.87	0.897	0.907	0.829	0.798



(a) UFC – ICA (b) UFC – Lasso (d) Wine – ICA (d) Wine – Lasso

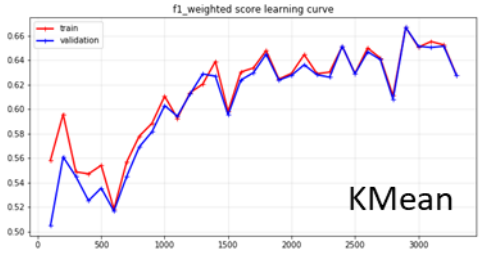
Figure 6. t-SNE Plots with KMeans Clustering with Reduced Dimension Data Set (ICA & Lasso)

6 NUERUAL NETWORK WITH ICA AND LASSO REDUCED FEATURES (WINE DATA SET)



With both ICA and Lasso, there is a notable reduction in computation time, a direct result of the smaller number of features that the network must process. This decrease in dimensionality means faster training and prediction times, which is beneficial for practical applications where speed is a concern. The ICA-reduced feature set seems to have a minimal impact on bias but significantly diminishes variance. This suggests that while the neural network's average performance across different datasets (bias) doesn't improve, its consistency does (variance). The lower variance indicates that the model is less prone to overfitting, possibly because ICA removes some noise and redundant information in the data, leading to more stable performance. In contrast, the Lasso model maintains a performance level close to that of the full dataset when used with neural networks, with only a modest decrease in accuracy (from 0.92 to 0.88). Given that the feature set was reduced to a third of the original size, this small loss in accuracy is noteworthy. It implies that the features retained by Lasso are highly predictive, encapsulating much of the information that the neural network needs to make accurate predictions. This efficiency makes Lasso an attractive option for feature selection when working with neural networks.

7 NUERUAL NETWORK WITH CLUSTERING LABELS (WINE DATA SET)



Labels from KMeans and EM used as input features simplify the feature space the neural network has to work with. This can lead to a more straightforward training process and faster convergence. The similarity in results from labels generated by both KMeans and EM suggests that the clusters are stable and capture fundamental characteristics of the datasets that are useful for classification tasks. Achieving an accuracy of 0.65 is notable, especially considering that these labels are unsupervised features. It indicates that the neural network can effectively use these cluster-derived features to make predictions, even if the clusters were not originally designed for classification. This approach could be likened to a form of ensemble learning, where the clustering algorithms perform an initial "vote" on the data's grouping, and then the neural network acts as a second layer to refine these results. This two-stage model can sometimes capture complex patterns more effectively than a single model working alone. The effectiveness of clustering labels as features suggests that they have captured significant information about the data's structure. This indicates potential for the clustering labels to be used as a kind of feature transformation, contributing to a neural network's ability to learn and make accurate predictions.