Unsupervised Learning and Dimensionality Reduction

Herbert Ssegane GtID#903755945

1.0 Introduction

This analysis explores differences and similarities among six unsupervised learning algorithms. Unsupervised learning focuses on extracting patterns from unlabeled data. Therefore, the learning algorithms learn without a teacher and thus focus on finding similarities among individual data samples given specific features. The two categories of unsupervised learning algorithms are 1) clustering; discovering unknown groups in data, and 2) dimension reduction; reducing the dimensions of the feature space to enable identification of unknown groups or to facilitate supervised learning without the curse of dimensionality. The curse of dimensionality refers to increased computational effort when working with high-dimensional data. The curse also refers to increased probability of introducing noise to the data as the dimensions of the feature space increases. The objective of this assignment is to compare six algorithms with emphasis on:

1) Behavior of two clustering algorithms on two datasets
2) Behavior of four dimension reduction algorithms on two datasets
3) Impact of dimension reduction on clustering outcomes
4) Impact of dimension reduction on performance of a supervised learning algorithm
5) Impact of clustering on performance of a supervised learning algorithm. Clustering on dimensionally reduced data.

These include two clustering algorithms and four dimension reduction algorithms. The two clustering algorithms are k-means (*k*-means) and expectation maximization (EM). The four dimension reduction algorithms are principal component analysis (PCA), independent component analysis (ICA), randomized projections (RP), and recursive feature elimination (RFE).

2.0 Data

The two datasets are the same datasets used in the supervised learning assignment. Below is a succinct description of the data and data preprocessing prior to implementation of the learning algorithms

2.1 Wisconsin breast cancer diagnostic data (WBCD)

The dataset consists of 569 data points with 30 features computed from each cancer cell nucleus to predict whether the cancer is benign (B) or malignant (M). For a detailed description of each feature, the reader is referred to *Kaggle (2022)*. The dataset is relevant because it is a real-world data of the second leading cause of cancer deaths in US women, at a fatal rate of 2.6% or 1 in 39 women [*Cancer, 2022*].

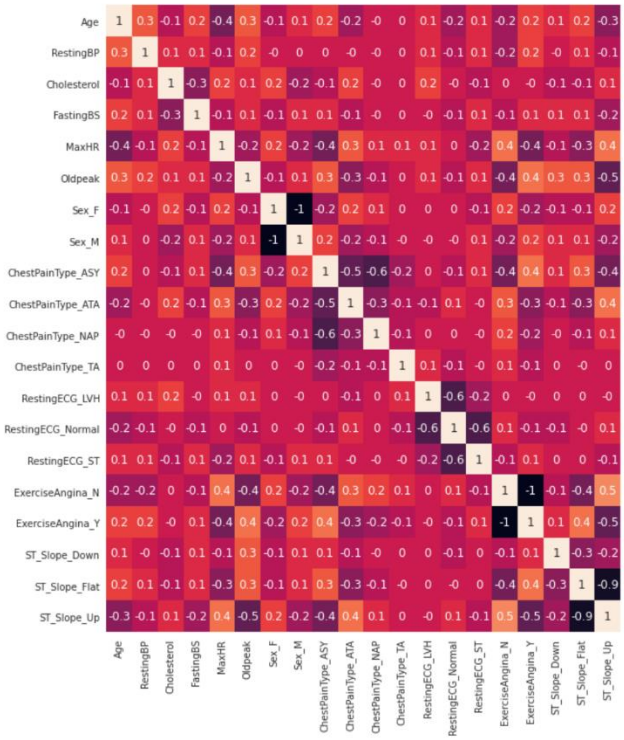
2.2 Heart failure prediction data (HFP)

The dataset consists of 918 data points with 11 features to predict heart disease where a value of one represents observed heart disease while a value of zero (0) represents no observed heart diseaseFor a detailed description of each feature and data sources, refer to *Fedesoriano (2022)*. This data is relevant because it demonstrates application of machine learning in the health care industry, knowing that heart failure is the leading cause of deaths, globally.

2.3 Data pre-processing

Both supervised and unsupervised learning algorithms expect numeric values. Thus, data preprocessing was implemented prior to exploring the learning algorithms. Refer to assignment 1 report for detailed pre-processing. To minimize the impact of differences in feature scales on some algorithms (e.g., kmeans), features were standardized such that their distribution had a mean of zero and a variance of one.

2.4 Data characteristics



This figure represents the pairwise correlation between features for the heart failure prediction (HFP) data. HFP has three feature pairs that are highly correlated (|corr| >= 0.9) , representing 15% of the feature space. The Wisconsin breast cancer diagnosis (WBCD) data, there are 15 highly correlated features, representing 50% of the feature space.

Individual feature variance (standardized features) for HFP varies between 0.02 and 0.25 with a mean of 0.14 while corresponding values for WBCD are 0.01, 0.02, and 0.05.

Figure 1: Correlation matrix of the heart failure prediction feature data.

3.0 Unsupervised learning

3.1 K-Means

K-Means is a hard clustering algorithm where the number of clusters (the *K* value) are pre-set such that the learning process finds the optimal cluster centers and assigns each data point to the nearest cluster center. The algorithm randomly initiates the centers and iterates the process of assigning points to the nearest center, then it recomputes cluster centers until the sum of squares (SS) and cluster centers are constant.

Approach: Scikit-learn python module's implementation of K-Means was used. The Euclidean distance metric was chosen because it captures the distance between any pair of points. For the two datasets, points close to each other infer similarity of features. Features were standardized using the MinMaxScaler (0, 1) because the Euclidean distance is not scale invariant. The mean Silhouette score of all data points was used to select the number of clusters. This score measures similarity of a data point to its cluster compared to other clusters (consistency within clusters). Yellowbrick python module was used to select *k*.

3.2 Expectation maximization (EM)

The EM is a soft clustering algorithm where each cluster is a generative model (e.g., Gaussian) defined by the mean (μ), standard deviation (σ), and a mixing coefficient (ρ). The algorithm returns the probability of data point belonging to cluster-k. The algorithm initiates parameters of distribution θ given k. It estimates values of unknown latent variables (Z) given θ (expectation step). Then it solves for new values of θ given Z (maximization step).

Approach: Scikit-learn python module's implementation of gaussian mixed models (GMM) was used. Number of clusters were estimated using Akaike's and Bayesian information criteria (AIC and BIC) to score the log-likelihood and complexity of the resulting GMM given k components (gaussian distributions). Lower AIC and BIC scores are preferred and therefore, the score with least number of components was used to estimate k (mostly BIC).

3.3 Principal component analysis (PCA)

PCA transforms high-dimension data into best low-dimension data such that the new data still captures most of the variability in the original data. The new features are referred to as components. They are linear combinations of features in original data, that are not correlated yet ordered such that first few components capture most of the original data variability.

Approach: Scikit-learn python module was used to implement PCA on two datasets. Used covariance matrix to generate the components because all features were standardized and thus on the same scale. Charts were generated to depict the cumulative explained variance as function of number of components. Corresponding eigen values were not plotted because they did not provide additional information.

3.4 Independent component analysis (ICA)

ICA transforms high-dimension data into low-dimension data by generating components that are independent. This is archived by minimizing the second, third, and fourth order moments in the data. Compare to PCA that generates uncorrelated components by exploring only the second order data moments. ICA assumes that the components are non-gaussian.

Approach: Scikit-learn python module was used to implement ICA on two datasets. The mean absolute kurtosis of k-components was used to select k. Higher kurtosis values represent non-gaussian distribution and are preferred.

3.5 Gaussian random projection (GRP)

GRP projects high-dimension data (d-dimension) to low-dimension data (k-dimension) using a random dimension matrix (k x d) such that $k << d$. The random matrix is generated using a gaussian distribution. The main advantage of this approach is the low computational effort, and it is robust to outliers.

Approach: Scikit-learn python module's implementation of random gaussian projections was used. Average correlation between reconstructed data and original data was used to score efficient number of projections (components).

3.6 Recursive feature elimination (RFE)

RFE is a wrapper approach to feature selection because it uses a supervised learning algorithm to assess performance of selected feature subsets. It selects feature subset by removing one feature at a time until optimal features remain.

Approach: Scikit-learn python module's implementation of RFE was used. For this analysis a boosting supervised algorithm was used as the model for the wrapping process while varying number of

selected features. Classification accuracy was used to quantify performance of selected feature subset.

4.0 Results

4.1 Behavior of two clustering algorithms on two datasets

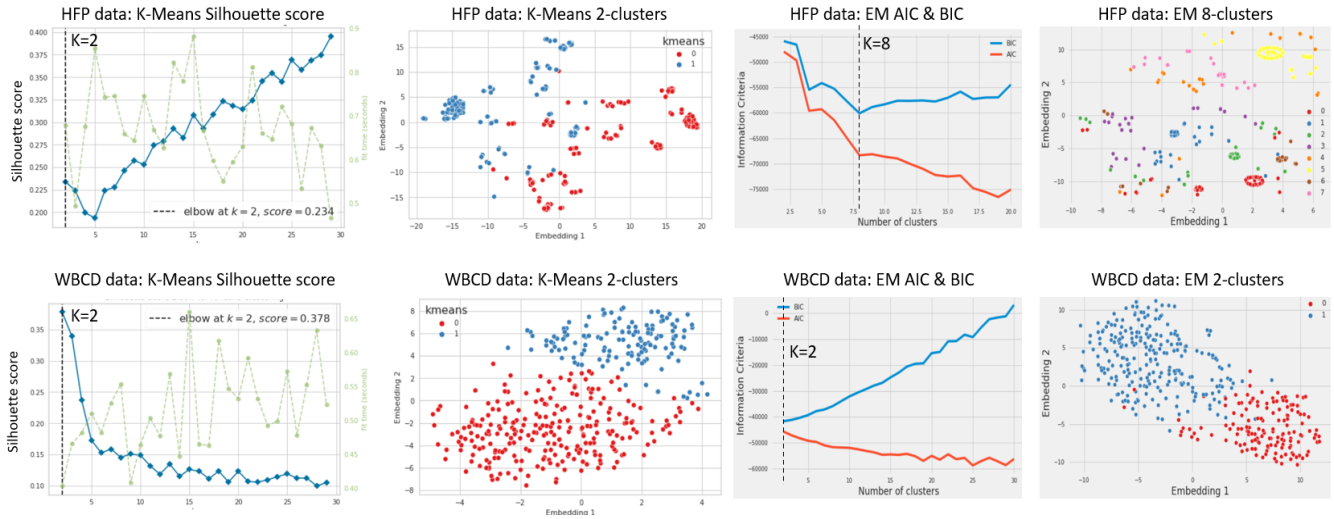


Figure 2: comparison of K-Means and expectation maximization (EM) clustering algorithms on heart failure prediction (HFP) and Wisconsin breast cancer diagnosis (WBCD) data

Observation: For the HFP and WBCD data, K-means decomposes the data to only 2 optimal clusters using the Silhouette score. Cluster visualization is based on PCA and t-distributed stochastic neighbor embedding (t-SNE), 2-D representation and assigning the corresponding cluster group to each data point. One can infer a linear separator with noise for the K-Means on both data. The HFP has more mis-classifications after applying a linear separator. Number of clusters for the EM are chosen using the AIC and BIC scores. Although, both AIC and BIC are plotted, BIC consistently generated the least optimal number of clusters and thus was used in subsequent analysis. K=8 is the lowest BIC for the HFP and k=2 is the lowest for WBCD. EM result is consistent with K-Means on WBCD but different on HFP. Both data are binary classification and thus 2 groups are expected. K-Means was consistent with expected response value (0 or 1) for both data but EM only consistent on WBCD. 87% - 92% points assigned to positive cluster by K-Means matched the positive disease classification. Five of eight clusters by EM on HFP can be reclassified as positive with 62% – 91% accuracy. Thus, EM is capturing more cohorts within the positive group.

Why? There are three major differences between HFP and WBCD. WBCD has more features (30) than HFP (20). However, HFP has less correlated features (15%) compared to WBCD (50%) – refer to section 2.4. The range of the variances of standardized features is larger for HFP (0.02 – 0.25) than WBCD (0.01 – 0.05). While K-means only uses the mean to update cluster centroids, EM uses both the mean and variance (covariance). The significant differences in variances of the two data partly explains the results of EM compared to K-Means. Choice of different metrics to quantify the quality of the clusters may yield different results. Thus, exploration of different metrics is the additional work not performed by this analysis.

4.2 Behavior of four dimension reduction algorithms on two datasets

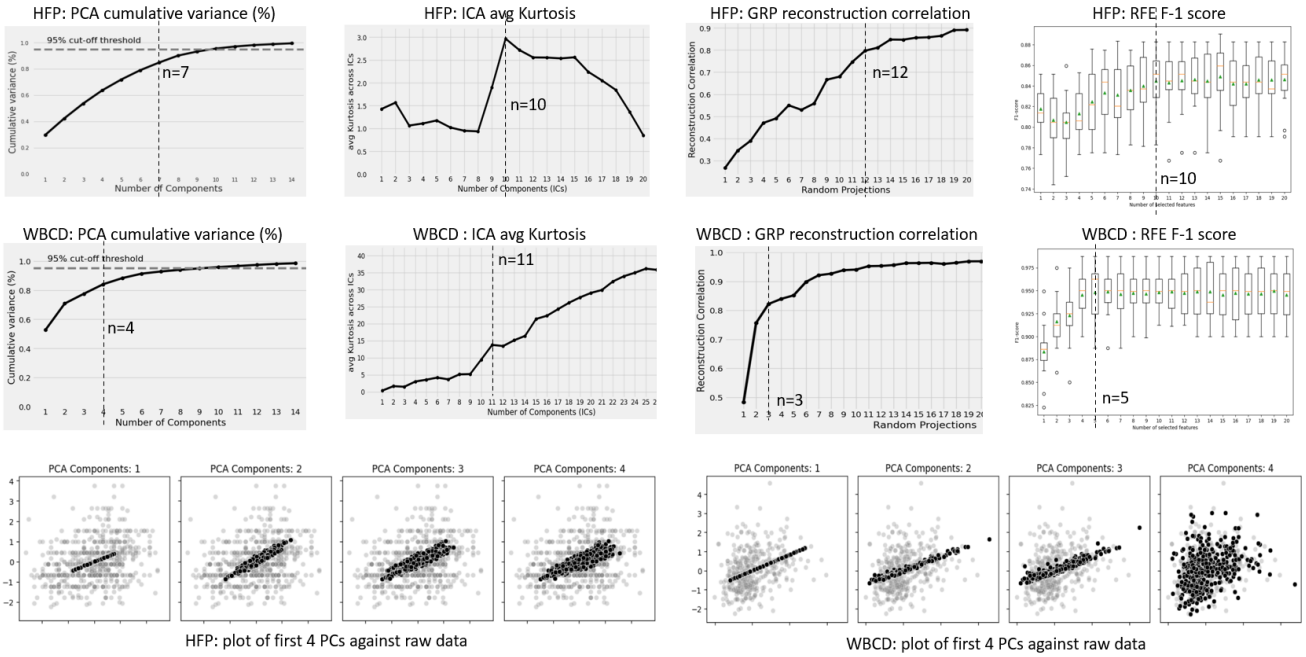


Figure 3: comparison of PCA, ICA, GRP, and RFE on heart failure prediction (HFP) and Wisconsin breast cancer diagnosis (WBCD) data. The figure also depicts the first 4 principal components (PCs) of both data.

Observation and why? The ICA is consistent across the two data with 10-11 selected components. The PCA, GRP, and RFE select more components or features from data with higher feature variance and low highly correlated features (HFP). For HFP 7-12 features are selected (few highly correlated features) by the 3 feature reduction algorithms while 3-5 features are selected for WBCD (many highly correlated features). Refer to section 2.4 for details on data characteristics. ICA maximizes independence of transformed low-space data while PCA maximizes variance. Thus, PCA results are meaningful. However, not sure why ICA gives similar independent components on two data with significant difference in variability and level of correlated features.

The visualized PCs show the ordered nature of PCA results where the first PC captures most of the variability. PCs 3 and 4 for HFP still represent significant variation based on well defined projection direction and less spread of the data along the projection. However, there is a significant jump in the represented variability of PCs 3 and 4 for WBCD. This observation is related to number of correlated features between the two data.

4.3 Impact of dimension reduction on clustering outcomes

Observation and why? Worked with both data but for observations and analysis only focused on the WBCD data to compare impact of different dimension reduction on clustering results. One major observation is the reduction in computation effort. Computation effort reduced from ~1.0s using all features to <0.5s (compare training time in Figure 2 to time in Figure 4). This is one of key advantages of dimension reduction. This impact should be more pronounced in higher dimension feature spaces than data used in this analysis. The other advantage is reduction in feature redundancy. Also, due to relatively small data size (<1000 data samples) and low feature space (30-features), the expected improvement in computation speed for the GRP compared to other methods is not demonstrated.

The number of clusters on the original WBCD was k=2 for both K-Means and EM. PCA, GRP, and RFE reproduce similar results for both clustering algorithms. ICA generates more clusters, 5 by K-Means and 8 by EM. From Figure 3, ICA generated more components than the other methods, 11 by ICA vs. <=5 by other methods. This partly explains the more clusters. Also the assumption that each data point is a mixture of independent components may not be met by this data and thus significant difference in the output compared to original data.

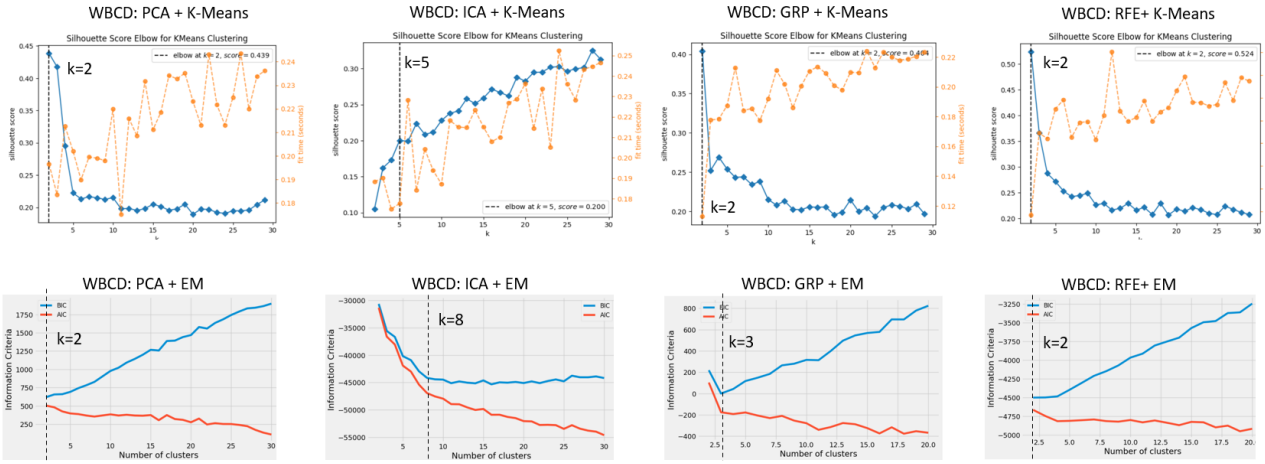


Figure 4: comparison of the impact of PCA, ICA, GRP, and RFE dimension reduction methods on the outcome of K-Means and EM clustering algorithms.

4.4 Impact of dimension reduction on performance of a supervised learning algorithm

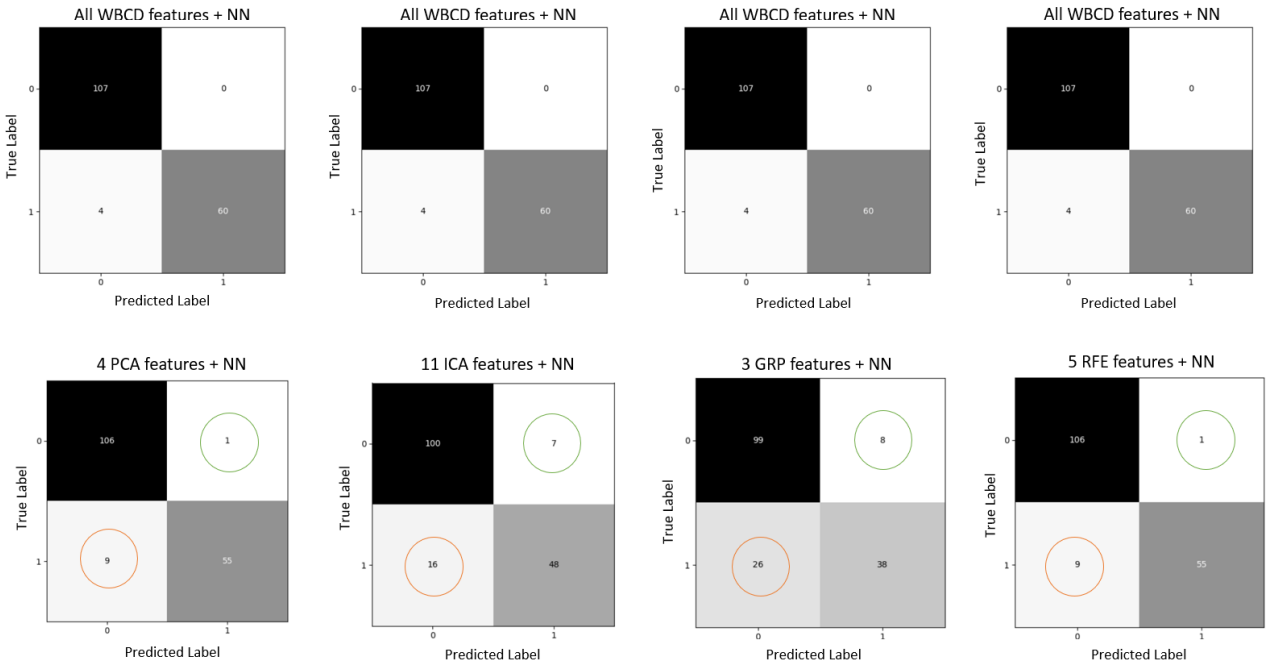


Figure 5: Classification confusion matrices to compare Neural Network (NN: MLPClassifier) performance on WBCD test data after PCA, ICA, GRP, and RFE feature transformation or selection. Same NN hyper-parameters were used for all classification scenarios.

Observation and why? None of the feature reduction and selection methods outperformed the original feature space on the classification of malignant (label=1) or benign (label=0) breast cancer. This is attributed to low dimension space for the WBCD data and thus it did not demonstrate improved model performance on lower dimension space. The reduced performance is attributed to increase in the false negative (true label=1 but predicted label=0) – orange circle (Figure 5) and increase in the false positives (true label=0 but predicted label=1) – green circle. PCA and RFE out-performed other methods. GRP had the least feature inputs (3) and the worst performance. The 4 PCA components capture >80% of the variability of the WBCD feature space, achieving an F1-score of 0.92 on the test data for the positive class. RFE used gradient boosting to select feature subset with high F1-score and thus, the selected 5 features achieved an F1-score of 0.92. For reference the F1-score on full features is 0.97.

4.5 Impact of clustering on performance of a supervised learning algorithm

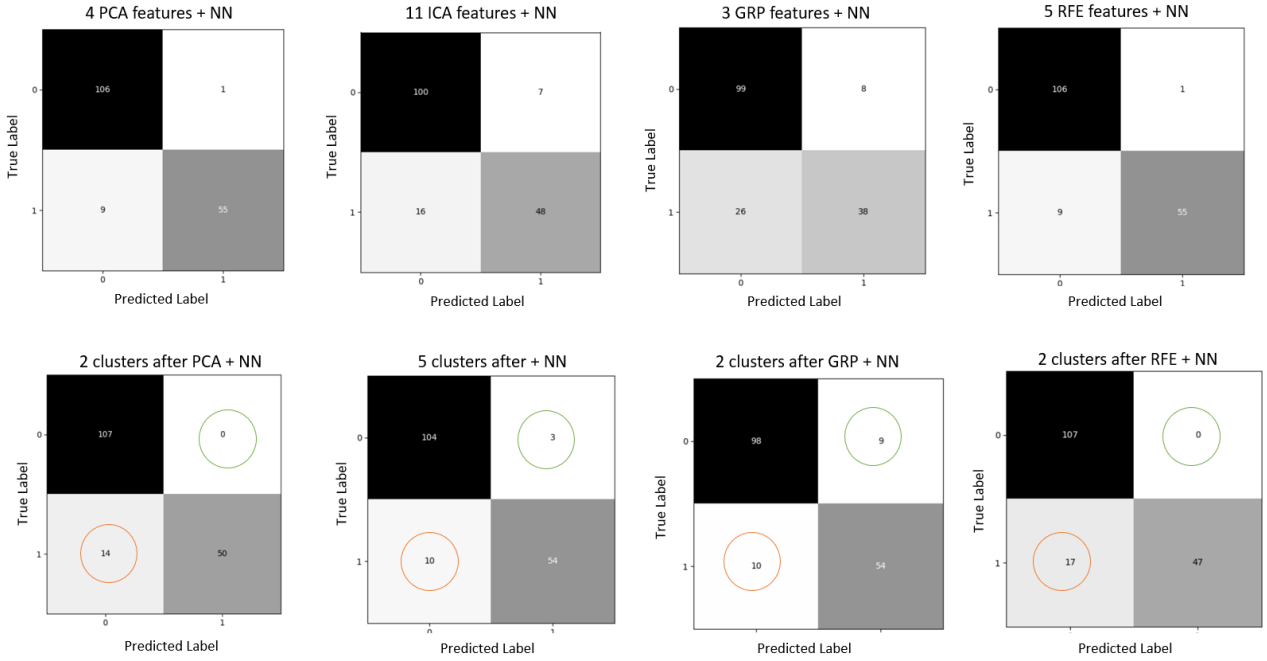


Figure 6: Classification confusion matrices to compare Neural Network (NN: MLPClassifier) performance on WBCD test data after clustering on PCA, ICA, GRP, and RFE transformed features. Same NN hyper-parameters were used in section 4.4 and 4.5.

Observation and why? Use of cluster classes after RFE feature selection had the highest increase in false positives (compare 5 RFE features + NN versus 2 clusters after RFE + NN). Thus, lowest NN classification performance. Also PCA + clustering + NN was worse than PCA + NN.

5.0 Conclusion

1. The first few principal components of PCA capture most of the original data variability. For both data, >80% of the variability is captured with <=7 first PCs. This is in line with the intrinsic ordering of PCA such that the 1st component captures the highest variability.

2. Other dimension reduction algorithms demonstrated similar results with high level of dimension reduction, thus, from 20-30 features to ~10 features.
3. Reconstruction correlation is a great measure of results from GRP. Number of projections were chosen where the reconstruction correction was >=0.8. This provides flexibility in choice of projections because you can set the desired correlation based on domain knowledge and original feature space.
4. Other than ICA, other dimension reduction methods were able to reproduce cluster groups same as original data
5. Neural network performance did not improve for PCA and RFE, but ICA and GRP benefited from clustering

References

1. Kaggle 2022. https://www.kaggle.com/uciml/breast-cancer-wisconsin-data, accessed 02-11-2022
2. Cancer 2022. https://www.cancer.org/cancer/breast-cancer/about/how-common-is-breast-cancer.html
3. Fedesoriano 2022. https://www.kaggle.com/fedesoriano/heart-failure-prediction