Unsupervised Learning and Dimensionality Reduction

Daniel Bach
dbach7@gatech.edu

1 DATASETS

1.1 Iris dataset

The iris dataset contains 150 instances with 4 features. Features included are the petal length, petal width, sepal length, and sepal width. Labels were encoded to be either 0, 1, or 2 indicating the species.

1.1.1 *Clustering hypothesis*

Clustering methods should work well on the iris dataset as there are only four features. Histograms of each feature show a Gaussian distribution for two of the four features.

1.1.2 *Dimensionality reduction hypothesis*

Dimensionality reduction would slightly improve the accuracy of models that use the transformed data as it should be able to capture collinearity between the features. For example, it is expected that an iris with bigger features will be on the higher end of the distribution in most measurements.

1.2 Wisconsin breast cancer dataset

The Wisconsin breast cancer data set contains 569 instances with 30 features. The features include numerous measurements of cell nuclei. Labels were encoded to be either 0, or 1 indicating the diagnosis.

1.2.1 *Clustering hypothesis*

Clustering method should not be as effective on the breast cancer dataset. This is due to the data having 30 features. The curse of dimensionality implies that the data points might be more sparse and therefore harder to cluster. Using all 30 features might also be troublesome as not all features might be equally as important so there is more potential for the data to be noisy.

1.2.2 Dimensionality reduction hypothesis

Dimensionality reduction techniques will greatly improve models that use the transformed data. Many features seem to be highly correlated, for example: mean radius, worst radius, radius error, etc. It should also be able to increase the accuracy of the clustering models since the number of dimensions are reduced, the data should become less sparse.

2 CLUSTERING

The two clustering algorithms used were the GaussianMixture model [1] and the AgglomerativeClustering model [2].

The GaussianMixture model uses the expectation maximization algorithm in order to assign data points probabilities of belonging to a cluster. The parameters passed into the model were the number of clusters. Means were also initialized for the model using the means of each feature for each label.

The AgglomerativeClustering model starts with each data point belonging to a cluster of itself and iteratively merges them together using a specified criteria. In this analysis, the

criteria was set to be the average of the euclidean distances. The parameters passed into the model were the number of clusters, meaning that the algorithm will stop once the data has been merged into this number of clusters.

For both clustering models, the accuracy of the model's labelings were calculated using the Adjusted Rand Index (ARI) ARI can range from -1 to 1, -1 meaning complete disagreement between cluster labels, 0 meaning no better than random labeling, and 1 meaning perfect agreement [3].

2.1 Iris dataset

2.1.1 *GaussianMixture model results*

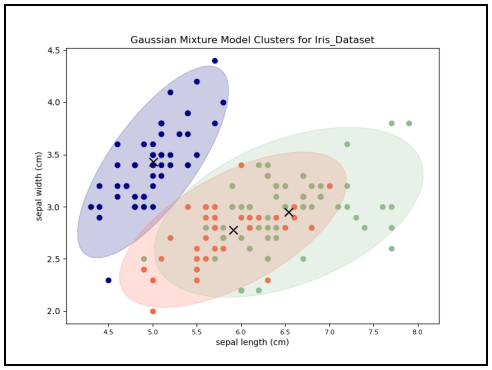

Figure 1 - GaussianMixture model clusters for the iris dataset

To more easily visualize the clusters, the plots were generated along just two of the four dimensions of the iris dataset. The GaussianMixture model was highly accurate in being able to label the data points. It achieved an ARI of 0.904. The time it took to fit the model was ~0.0075 seconds.

For the iris dataset, not initializing the means was also tested to see if it would have a significant impact on the ARI score: it did not. This could be due to the fact that by default, the GaussianMixture model uses K-Means to initialize the responsibilities for the expectation-maximization algorithm.

2.1.2 *AgglomerativeClustering model results*

The AgglomerativeClustering model did not perform as well as the GaussianMixture model as it achieved an ARI of 0.759 for the iris dataset. However, the algorithm was able to converge to a result much faster: ~0.00044 seconds. This might be due to the GaussianMixture model's expectation maximization algorithm taking longer to converge, which is affected by its *tol* parameter [1].

2.2 Breast cancer dataset

2.2.1 *GaussianMixture model results*

For the breast cancer dataset, the GaussianMixture model achieved an ARI of 0.701. The time it took to fit the model was ~0.1156 seconds. This is significantly longer than it took to train the GaussianMixture model for the iris dataset. This could be due to having more data points which contribute to a higher number of calculations in the expectation-maximization algorithm; there are also 26 more features than the iris dataset.

Unlike the iris results, when the means were initialized via K-Means instead of using the mean of every feature for the two labels, the ARI improved to 0.812. This would suggest that cluster centers from K-Means were better initial means than just simply taking the average of the features for each label.

2.2.2 *AgglomerativeClustering model results*

Table 1—ARI for different linkage criterions for the Wisconsin breast cancer dataset

Criterion	ARI Score
'average'	0.0523
'ward'	0.287
'complete'	0.0523
'single'	0.0024

AgglomerativeClustering performed poorly in its clustering of the breast cancer data. Changing the linkage criterion to 'ward' showed a significant improvement in the ARI. This is due to 'average' treating each dimension the same in terms of computing averages of the euclidean distances between clusters, while 'ward' aims to minimize the variance between two clusters to join them together [2]. 'single' performed the worst as an ARI score of 0 is the expected score with random labeling of data points. AgglomerativeClustering seems to be affected more by the number of features since euclidean distances are calculated for each data point then compared to created clusters. Dimensionality reduction should help to improve the performance of AgglomerativeClustering.

3 DIMENSIONALITY REDUCTION

Four dimensionality reduction techniques which were used on the datasets. They are principal component analysis (PCA), independent component analysis (ICA), random projection (RP), and multidimensional scaling (MDS).

The variance inflation factor (VIF) is a measure of the amount of multicollinearity [5] which might adversely affect regression models, leading to insatiable weight estimates, overfitting, and increased sensitivity to noise [6]. VIF before and after using the two dimensional reductions will be measured. VIF values greater than 10 represent high levels of multicollinearity between other features while a VIF of 1 represents no multicollinearity [6].

Table 2—VIF values of the original iris dataset

Feature	VIF
sepal length (cm)	263.0
petal length (cm)	173.0
sepal width (cm)	96.35
petal width(cm)	55.50

Table 3—VIF values of the original breast cancer dataset (4 highest values shown)

Feature	VIF
mean radius	63,306.
mean perimeter	58,123.
worst radius	9,674.7
worst perimeter	4,487.8

3.1 PCA

For PCA, the scikit-learn module allows users to specify how many principal components are created [4]. The range can be from [1, number of features]. *n_components* was set to be 0.95, so that the PCA module will create enough components such that 95% of the variance in the data is explained by the principal components. Each dataset was standardized before performing PCA such that dimensions with inherently larger magnitudes do not dominate the contributions to principal components [6].

3.1.1 VIF values after PCA

After running PCA on both datasets, the VIF for each principal component became 1.0. This is because PCA constructs linear combinations of the original features such that the components are orthogonal to each other [6].

3.1.2 Iris dataset

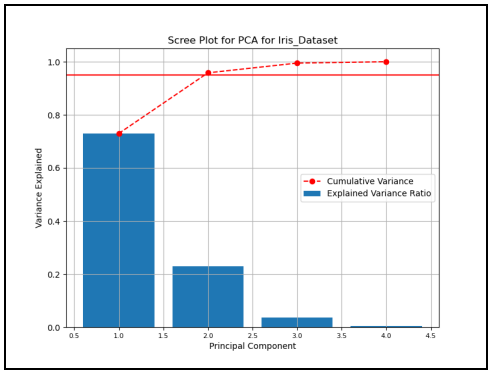

Figure 3 - Scree plot for the iris dataset

Scree plots show how much of variance of the original data set can be captured with an increasing number of principal components [6]. Two principal components are needed to explain ~95% of the variance in the Iris data. The first principal component accounts for a large portion as well, around 73%. The most important features which contributed the most to each principal component [7]. For the first principal component, the largest contribution comes from the petal length. For the second principal component, the largest contribution came from the sepal width.

3.1.3 Breast cancer dataset

PCA was able to capture 95% of the variance in the breast cancer data with just 10 components, compared to the original 30 features. The number of features was able to be greatly reduced since the magnitude of the VIF for the features were so high.

A few of the features with the largest contribution to the principal components were: mean concave points, mean fractal dimension, texture error, and worst texture. All ten of the most important features can be viewed via the code.

3.2 ICA

For ICA, scikit-learn's FastICA module was used. VIF values for every independent component for both datasets were 1.0. This indicates that the independent components were statistically uncorrelated.

3.2.1 Iris dataset

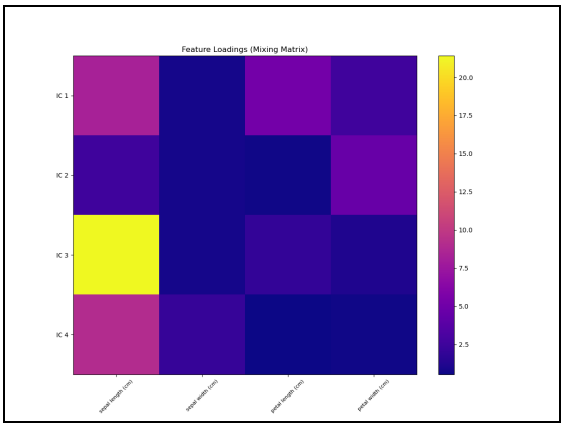

Figure 4 - ICA mixing matrix for iris dataset

The mixing matrix generated for the iris dataset shows that for all independent components, the sepal length has a larger coefficient than all other features, which is different from PCA.

Since ICA is most effective when the data is non-gaussian [6], ICA won't be as effective on the iris data as a plot of the kurtosis

values shows that the iris data are close to a Gaussian distribution.

3.2.1 Breast cancer dataset

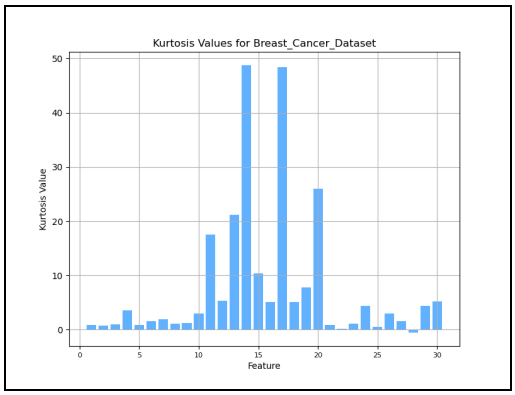

Figure 5 - Kurtosis values plot for each independent component

ICA is much more effective on the breast cancer dataset since many features deviate from a normal Gaussian distribution (described when kurtosis value = 3) [6].

3.3 RP

RandomProjection is faster computationally than the other three algorithms in this section as it projects high dimensional data into a smaller dimensional space using a random matrix filled with values from a gaussian distribution [8]. For random projection, experiments focused on how well RP could reproduce the original datasets.

3.3.1 Reconstructing data from RP

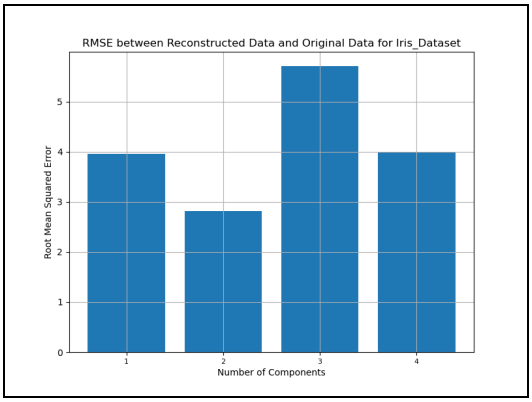

Figure 6 - RMSE from a reconstructed Iris dataset from RP

Initially, it was thought that with a greater number of projections the original data could be better reconstructed. However, the reconstructed iris and breast cancer data sets showed no pattern of decreasing RMSE with higher number of projections.

3.4 MDS

For MDS, the scikit-learn module allows users to specify how many dimensions to reduce the data to via the *n_components* parameter [9]. For this analysis, it was left to the default of 2 for ease of visualization.
While MDS's dimensions both had much lower VIF values for each dimension than the original dataset, they are not equal to 1.0. This is because scikit-learn's MDS uses a non-metric algorithm, resulting in dimensions that are not necessarily orthogonal [6].

3.4.1 Iris dataset

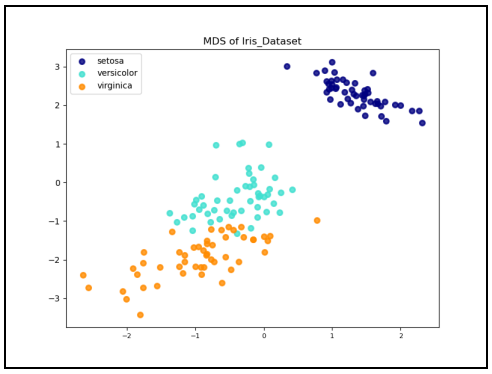

Figure 6 **-** Iris dataset reduced to 2-dimensions after MDS

By inspecting the visualization of the transformed iris dataset, it is shown that the clusters are more easily discernible. This should lead to improvements in both the clustering models as there is less intermixing of labels in the reduced MDS space. Both the dimensions in the MDS have a VIF of 4.422 which is lower than the original VIF for every feature in the iris dataset.

3.4.2 Breast cancer dataset

When the breast cancer data is reduced to 2-dimensions via MDS, it's clear that there is still room for improvement as the region around (0, 0) shows a lot of intermixing between data points of different labels. The VIF measured for the two dimensions is 4.206 which is around 3 or 4 magnitudes (10^3 ~ 10^4) smaller than the dimensions were originally.

4 CLUSTERING REDUCED DATA

4.1 GaussianClustering on PCA

Table 4 —ARI scores for PCA reduced datasets

Dataset	**Adjusted Rand Index (ARI**
Breast cancer	0.137
Iris	0.727

The ARI for breast cancer decreased from 0.701 to 0.137. The ARI for iris decreased from 0.904 to 0.727. The reasoning for this loss can be due to several reasons. One is that when PCA was performed, only enough principal components were kept to explain 95% of the variance in the data. Here, the GaussianClustering model had less information than it did before, when it was trained on all the original data. It was expected that GaussianClustering would perform better after PCA, since the data was standardized before dimensionality reduction.

4.2 GaussianClustering on MDS

Table 5 —ARI scores for MDS reduced datasets

Dataset	**Adjusted Rand Index (ARI**
Breast cancer	0.682
Iris	0.886

The ARI for breast cancer decreased slightly from 0.701 to 0.682. The ARI for iris also decreased slightly from 0.904 to 0.886. These decreases are a lot smaller than the results for the PCA reduced data. This is surprising as MDS reduced the data to two dimensions for both data sets, whereas PCA had 10 principal components for the breast cancer data and 2 for the iris data.

5 NEURAL NETWORK WITH DIMENSIONALLY-REDUCED DATA

In this section, the MLPClassifier neural network will be trained on data that has been dimensionally-reduced via ICA and MDS.

The MLPClassifier with the best parameters from assignment 1 achieved a 93.3% test accuracy.

5.1 MLPClassifier with RP data

The MLPClassifier was able to achieve 100.0% test accuracy on a subset consisting of 20% of the data.

5.2 MLPClassifier with MDS data

The MLPClassifier was able to achieve 96.7% test accuracy on a subset consisting of 20% of the data.

6 NEURAL NETWORK TRAINED WITH ADDITIONAL CLUSTER DATA

In this section, the original iris dataset will contain one additional feature: the cluster labeling from the GaussianClustering model and the AgglomerativeClustering model. The MLPClassifier will be trained on these two 'new' datasets and scored on test accuracy.

6.1 Results

Table 6 —Test scores for MLPClassifier

Clustering Model	Test Set Score
GaussianClustering	100.0%
AgglomerativeClusterin	96.7%

7 CONCLUSION

GaussianClustering outperforms AgglomerativeClustering on both the iris and breast cancer dataset.

Some information is lost when performing dimensionality reduction but having less features reduces the noise that is fed into the models. Dimensionality reduction greatly reduces the VIF, which will become 1 if components / projections are orthogonal.

Even though the data appeared more separable visually, the ARI score for the clustering models dropped for both dimensionally-reduced datasets.

The MLPClassifier was able to outperform the previous MLPClassifier trained on 80% of the original training data when trained with 80% of the dimensionally reduced (via RP and MDS) data.

By introducing a new feature indicating the labeling given by the clustering algorithms to the iris dataset, the MLPClassifier was able to improve its test accuracy.

8 REFERENCES

1. https://scikit-learn.org/stable/modules/generated/sklearn.mixture.GaussianMixture.html#sklearn.mixture.GaussianMixture
2. https://scikit-learn.org/stable/modules/generated/sklearn.cluster.AgglomerativeClustering.html#sklearn.cluster.AgglomerativeClustering
3. https://scikit-learn.org/stable/modules/generated/sklearn.metrics.adjusted_rand_score.html
4. https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.PCA.html

5. https://www.investopedia.com/terms/v/variance-inflation-factor.asp
6. https://chat.openai.com/share/b1916ffc-4bbe-4cc9-821b-32f33ef28552
7. https://stackoverflow.com/a/50845697/22862849
8. https://scikit-learn.org/stable/modules/generated/sklearn.random_projection.GaussianRandomProjection.html#sklearn.random_projection.GaussianRandomProjection
9. https://scikit-learn.org/stable/modules/generated/sklearn.manifold.MDS.html#sklearn.manifold.MDS