Comparing PCA and Neural Encodings on Spectral Clustering

Will (James) Stonebridge [1]

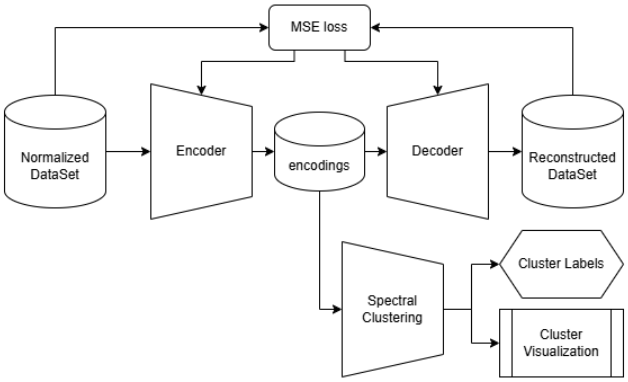


Figure 1. The Neural+Spectral Algorithm described in section 2.2

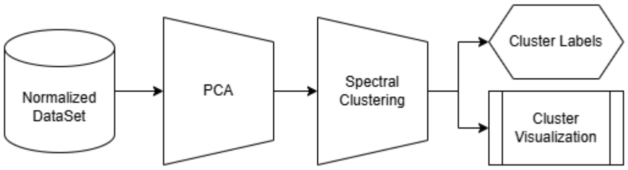


Figure 2. The PCA+Spectral Algorithm described in section 2.2

1. Problem Statement

In Clustering Problems, we seek to identify groups of similar data points within large datasets. When using datasets with less than 3 dimensions, finding these groups becomes significantly easier for two reasons:

- Datasets can be visualized. This gives the Data Scientist invaluable qualitative information about the size, shape and relative position of any clusters.
- The curse of dimensionality does not apply at less than 3 dimensions. In contrast, clustering algorithms struggle with data that is highly dimensional due to the failure of distance metrics to differentiate points.

Unfortunately, almost all real world datasets possess more than 3 features. Consequentially, there is a strong incentive to find effective ways to reduce dataset dimensionality while retaining enough information to distinguish the clusters in the data.

In this paper, I compare two algorithms that solve this problem: Neural+Spectral and PCA+Spectral. Both algorithms take in highly dimensional datasets and produce cluster labels and 3D cluster visualizations. The algorithms are evaluated across 4 different datasets on 4 metrics representing cluster cohesiveness, separation and correctness. Finally, the resulting metrics are discussed and analyzed.

2. Algorithms

Both of the following algorithms consist follow the same basic pattern: The dataset is normalized, the dimensionality of the dataset is reduced, and, finally, Spectral Clustering is applied to the reduced data points. Since both dimensionality reduction methods are negatively impacted by features of different scales, normalization was a critical step. Additionally, it was found that both dimensionality reduction methods produced non-circular, non-uniform, but connected clusters. The non-linearity and differing densities matched the strengths of Spectral Clustering, making it the obvious clustering algorithm choice. The Spectral Clustering parameter n is chosen based on a category feature specific to each dataset.

2.1. Nueral+Spectral

Neural+Spectral utilizes a Linear Neural AutoEncoder to acquire a 3D representation of the dataset. As shown in figure 1, A dataset of dimensionality k is fed into a encoder composed of fully connected layers ($k \Rightarrow 10 \Rightarrow 5 \Rightarrow 3$) which produces a 3D representation of the dataset. This encoding is then fed into a decoder of fully connected layers ($3 \Rightarrow 5 \Rightarrow 10 \Rightarrow k$), which produces a recreation of the dataset in its original dimensionality. The dataset is trained on the Mean Squared Error Loss between the original dataset and it's recreation.

2.2. PCA+Spectral

PCA+Spectral utilizes PCA to acquire a 3D representation of the dataset. The covariance matrix of the dataset is found. The dataset is then reduced using the 3 eigenvectors corresponding to the greatest eigenvalues of the covariance

Dataset	Distinct Categories	#features	#Entries	#epochs	batch size	Final Autoencoder MSE
Wine	2	12	4898	5	16	0.7392
Cancer	2	30	569	15	8	0.4323
Images	6	19	2310	12	8	0.8399
Glass	7	9	214	20	4	0.9338

Table 1. Basic Statistics corresponding to each Dataset used. The first column names the dataset. The next three columns describe the dataset itself. The next two columns describe the hyperparameters used when the autoencoder in Neural+Spectral was trained on this dataset. The final column is the final loss of the aforementioned autoencoder.

matrix.

2.3. Spectral

As a baseline, Spectral Clustering is also used with the same normalized dataset.

3. (Bonus) Datasets

All 4 Datasets were acquired from UC Irvine's Machine Learning Repository (http://archive.ics.uci.edu/).

Wine: Represents about 5000 different variants of ”Vinho Verde” wine. Categories in this dataset are defined as whether the wine is Red or White. This is the ”easy” dataset as it has a high number of entries, a reasonable initial dimensionality and only 2 categories to identify. It should also be noted that this data set possessed a much larger amount of White wines (3x the number of entries of red). Because of the large amount of data, the autoencoder converges at only 5 epochs, even with a larger batch size.

Cancer: Represents about 600 breast cancer tumors. Each Tumor is either malignant or benign. Notably, this dataset has a high number of features (30) while possessing only a few hundred entries. Due to the small number of entries, the autoencoder trained on this data for more epochs (15) and with a small batch size (8).

Images : This dataset represents the outputs of a neural segmentation model on about 2000 images. The 6 categories in the dataset represent a image subject (a patio or a brick wall, for example). Each feature is the neural embedding value output by the neural segmentation model when given the image. This dataset has more categories, but a larger number of entries relative to Cancer. To compensate for the higher dimensionality, we train across 12 epochs with a batch size of 12.

Glass: This dataset represents the composition of about 200 glass shards. With 7 categories and only 200 entries, it is by far the most difficult set to cluster. Due, to the limited amount of entries we chose to train over 20 epochs and with a relatively small batch size of 4.

3.1. Metrics

To evaluate the performance of these algorithms, three unsupervised metrics and one supervised metric is employed.

Silhouette Score: Measures the cohesiveness and separation between each labeled cluster. (-1 indicates poor clustering and 1 indicates perfect clustering)

Davies Bouldin Score: Measures the ratio of intra-cluster dispersion to inter-cluster distance. (Scores closer to zero indicate compact and separate clusters).

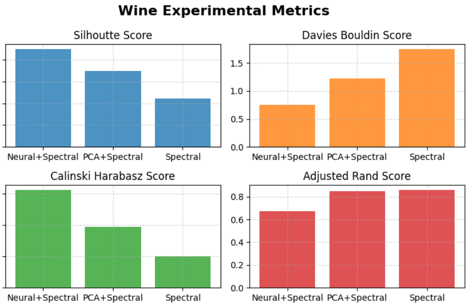
Calinski Harabasz Score: Measures the ratio of within-cluster variance to the variance between different clusters. (Higher Scores indicate well-defined clusters).

Adjusted Rand Score: Evaluates how much predicted clustering labels match a ”Ground Truth” set of clustering labels. In our case the ground truth labels are the categories in Table 1 (-1 indicates poor clustering and 1 indicates perfect clustering).

4. Experiments

All experiments follow the same basic procedure. Either PCA or an AutoEncoder (pretrained using the parameters described in Section 3) reduces the dataset to 3 principal components/encodings. These new data points are then labeled by the spectral clustering algorithm.

4.1. Wine



Both algorithms perform extremely well on this dataset. In particular, Neural+Spectral outperforms the other two

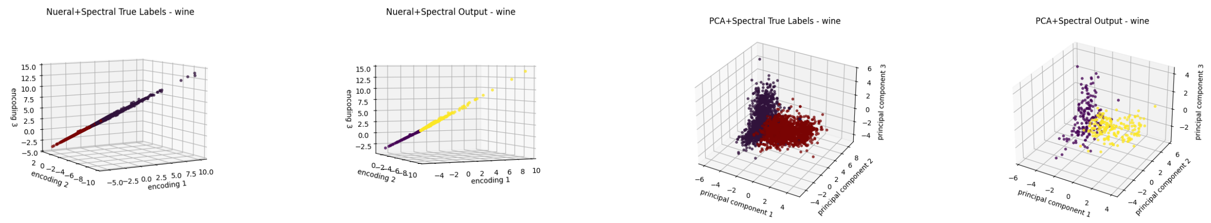
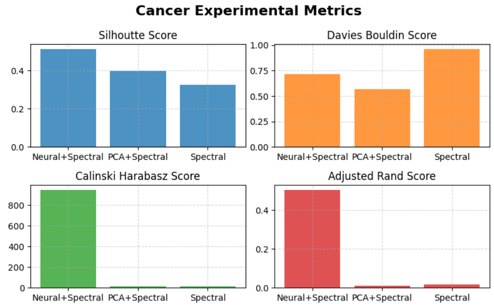


Figure 3. From left to right: Autoencoder output colored according to the Red/White Wine labels provided in the dataset, Autoencoder output colored according to the labels found by Spectral Clustering, PCA output colored according to the Red/White Wine labels provided in the dataset, PCA output colored according to the labels found by Spectral Clustering

algorithms in all 3 unsupervised metrics. Looking at Figure 3, this makes sense given that the Encodings are much more compact and separate than the PCA output. Nonetheless, PCA+Spectral and simple Spectral outperform Neural+Spectral on the Adjusted Rand Score. This could be because PCA+Spectral and Spectral retained more distinguishing information (their shapes are more 3 dimensional, whereas Neural+Spectral's dataset is very linear).

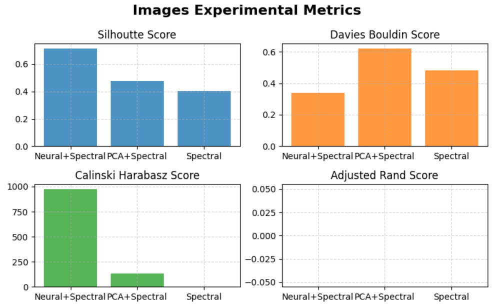
4.2. Cancer

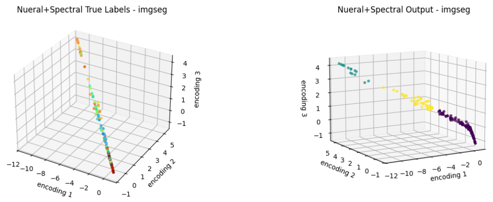


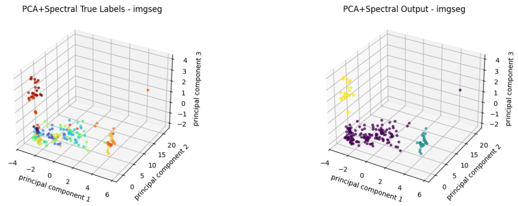
Neural+Spectral outperforms PCA+Spectral in all but one metric on this dataset. Looking at Figure 4, it is not hard to see why. It appears that Spectral clustering completely fails to find any useful clustering on PCA's output. It is also interesting to note that the Neural encodings appear to have identified 6 different vectors representing the data, potentially indicating there are 4 more sub categories in the data.

Lastly, it should be noted that Spectral performs the worst on all 4 metrics. This is likely due to the curse of dimensionality, which is particularly present in this 30-dimensional dataset.

4.3. Image Segmentation







As the above figures demonstrate there appear to be 3 distinct clusters in the 3d space of this dataset. However, the provided categories of this dataset do not at all represent those three colors (both "True labels" figures look like rainbows). Consequentially we change the n parameter of spectral clustering to 3.

After doing this it appears that Neural+Spectral vastly outperforms PCA+Spectral and Spectral on the 3 remaining unsupervised metrics. However, a visual inspection of the cluster graphs reveals that PCA+Spectral's output is very reasonable.

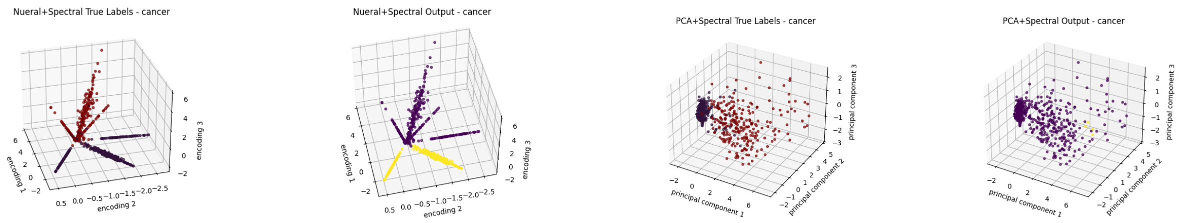
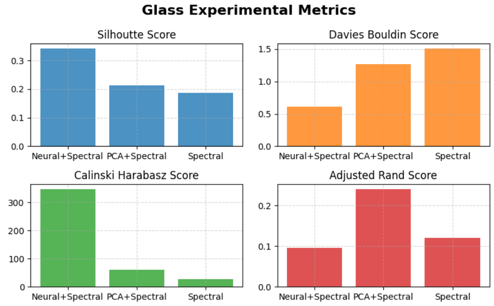
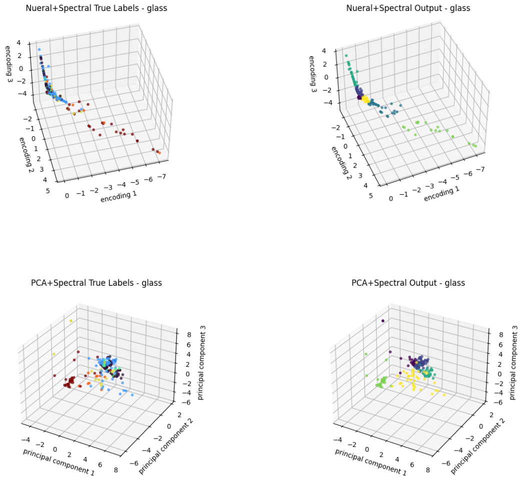


Figure 4. From left to right: Autoencoder output colored according to the Malignant/Benign labels provided in the dataset, Autoencoder output colored according to the labels found by Spectral Clustering, PCA output colored according to the Malignant/Benign labels provided in the dataset, PCA output colored according to the labels found by Spectral Clustering

4.4. Glass





As with the previous dataset, the provided ”True labels” do not appear to be very helpful in identifying any meaningful clusters.

It appears that Neural+Spectral slightly under performs in the Supervised Adjusted Rand Score. A possible cause of this failure could be that the training of this autoencoder was extremely unstable. I suspect that if this dataset had more entries (or few labels), Neural+Spectral would perform better on the Adjusted Rand Index.