Case Study – I: Model selection for Clustering

AAYUSH NAGPAL: 2753386N

HAET NIRAV TRIVEDI: 2771354T

KANHA BANSAL: 2789927B

TARITRO GHOSHAL: 2776608G

Introduction

PathalogyGAN is a dataset that consists of colorectal tissue data from 5000 images in form of vectors. Each vector in this set is of 200 dimensions and consists of 5000 such records. For this study, the dataset was reduced using two different algorithms for dimensionality reduction, namely- PCA and UMAP. These methods perform dimensionality reduction, and the resultant is a set of 5000 vectors with 100 dimensions from each algorithm. The reduced data from both algorithms are stored in a h5py file. **ResNet50** also contains the data of the same images in form of vectors, but the dimension of each vector is 2048. The same dimensionality reduction process is applied to ResNet50 and the resultant 5000 vectors of 100-d each are stored in h5py format. will perform K-means clustering and Hierarchical clustering on both the UMAP and PCA projections of the two representations. The focus of this case study is to build clustering by tuning its hyper parameters using appropriate methods and analysing the performance of these models by checking the cluster quality using intrinsic and extrinsic measures. To evaluate these models Silhouette Score, V-Measure, Bouldin-Davies Score, and Calinski-Harabasz index are used.

Methodology

K-Means is a popular unsupervised clustering algorithm. The aim is to divide n vector quantities into k clusters. It divides the data into k clusters and assign each data point randomly to a cluster. Then compute the centroid of all the clusters, reassign the point to the cluster with closest centroid. This process continues until points are no longer changing their positions/ clusters. To determine the optimal k-value (number of clusters) we can use a variety of methods like elbow method and silhouette analysis. Using the elbow method, we plot a graph between WCSS score (sum of squared distances of samples to their closest cluster centre, weighted by the sample weights if provided.) against the corresponding k value. By looking at the elbow point of the resultant graph we can find the optimal number of clusters (Refer Fig 1).

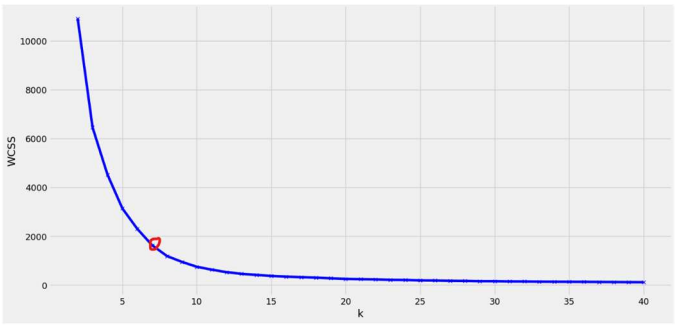

Fig 1. Elbow Method Graph

Another method to find optimal k-value is via silhouette analysis. For set of possible k values evaluate the silhouette score. The highest silhouette score will give us the optimal value of k. Fig 2 shows an example of silhouette analysis.

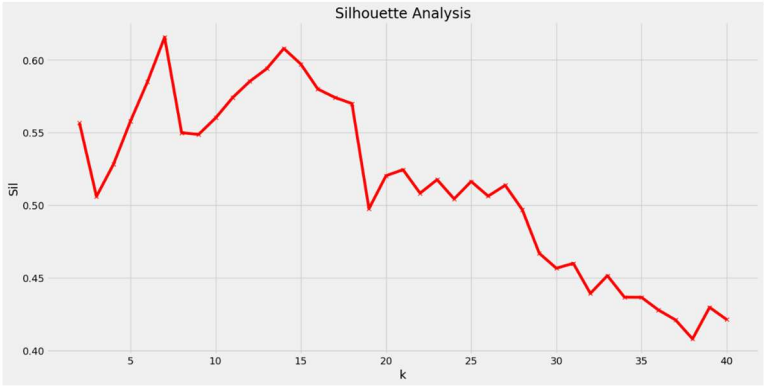

Fig 2. Silhouette Analysis

Hierarchical Clustering is also an unsupervised clustering algorithm. It forms distinct clusters but the objects within the clusters are similar. For this study, we have adopted Agglomerative Nesting approach which is a bottom-up approach. In this, each object forms its own cluster and then pairs with a similar cluster as we go up. This pairing of clusters continues till total cluster count is 1. This process can be plotted using a Dendrogram which shows Euclidean Distance between cluster points on x and y axis. On drawing a horizontal straight line along x axis, the number of clustering lines that intersect this horizontal line will give us the optimal number of clusters.

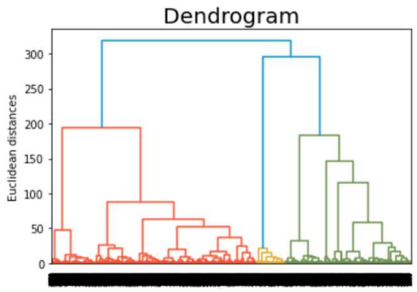

Fig 3. Dendrogram for a set of data points

The performance measures used in the experiment are explained below: -

- **Silhouette Score –** This score tells the goodness of clustering. It ranges between [-1,1]. 1 Means that the distance between clusters is significant. If the score is 0, then the distance is insignificant and negative score means that the clustering is wrong. The higher the score is the better.
- **V-Measure –** Based on the definition used in scikit-learn documentation V-Measure score is harmonic mean of the completeness of clustering and homogeneity. It ranges between [0,1].
- **Davies-Bouldin Score –** SciKit documentation defines DB scores as "average similarity measure of each cluster with its most similar cluster, where similarity is the ratio of within-cluster distances to between-cluster distances". Thus, clusters which are farther apart and less dispersed will result in a better score. Lower score means better clustering.
- **Calinski-Harabasz Score –** SciKit documentation defines this as "ratio of the sum of between-cluster dispersion and of within-cluster dispersion". The score is higher when clusters are dense and well separated, which relates to a standard concept of a cluster.

Experimental Setup

In this section, we will discuss the experimental setup and tuning of hyperparameters for both the representations.

PathalogyGAN representation

As mentioned earlier for every representation we will test both UMAP and PCA projection using Kmeans and Hierarchical Clustering. Using the h5py file, we will extract UMAP and PCA features from it. Using the UMAP features, we will evaluate the WCSS score and silhouette score corresponding to a range of k value (between 2 to 40). Using this we can generate the graphs as described and above perform silhouette analysis and elbow test which will give us the optimal value for k. In this case, the optimal number of clusters is 7. As shown in Fig 3(B), the silhouette score is maximised at k=7 and that is the optimal number for clusters for UMAP features.

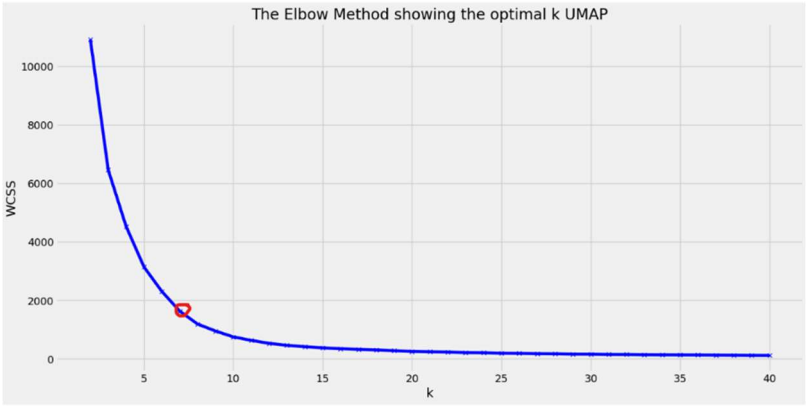

Fig 3. (A) Elbow Test for UMAP Features

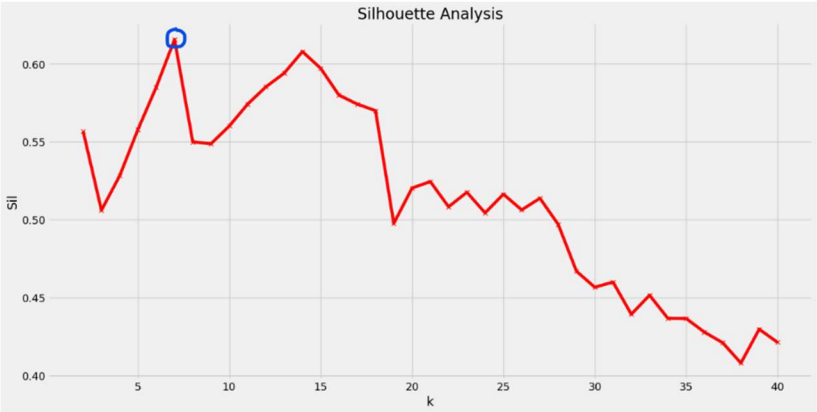

Fig 3. (B) Silhouette Analysis for UMAP Features

The same process is applied for the PCA features. As observed in Fig 4(A), the elbow test does not yield a clear breakpoint but in Fig 4(B) the silhouette score is maximised at k=2. Therefore, the optimal number of clusters for clustering PCA features using Kmeans is 2.

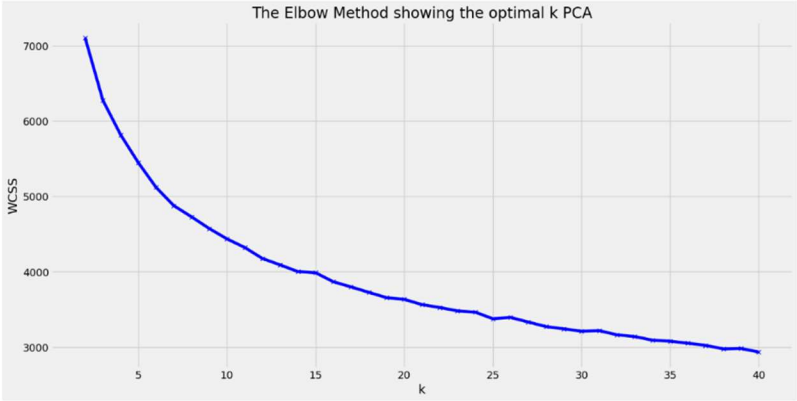

Fig 4. (A) Elbow test for PCA features

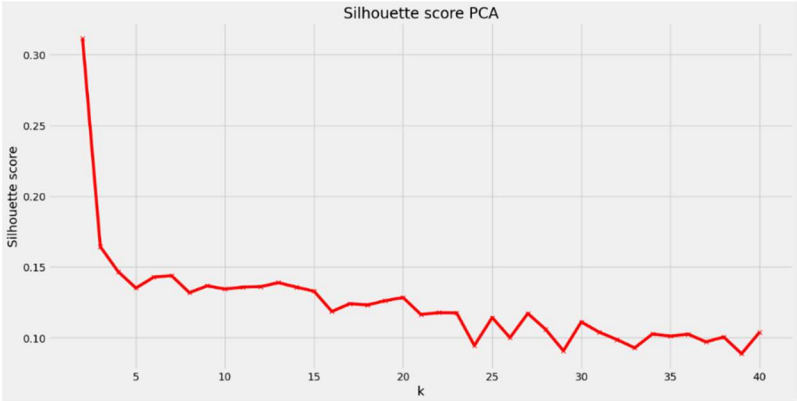

Fig 4. (B) Silhouette Analysis for PCA features

The second clustering algorithm applied is Hierarchical Clustering using Agglomerative Nesting. In this case as well we need to find the optimal number of clusters. The graphical method to find the optimal

value is a dendrogram. For PCA features the ideal number of clusters is 3 and for UMAP features it is 7 as shown in dendrograms in Fig 5.

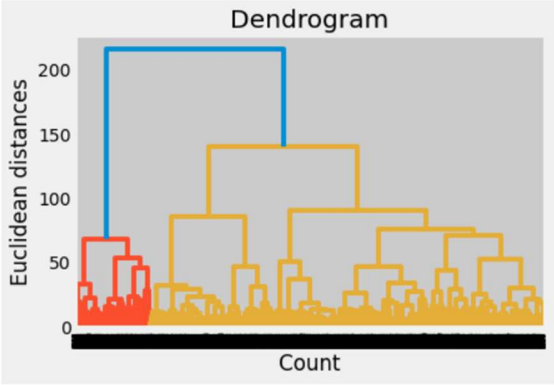

Fig 5. (A) Dendrogram for PCA

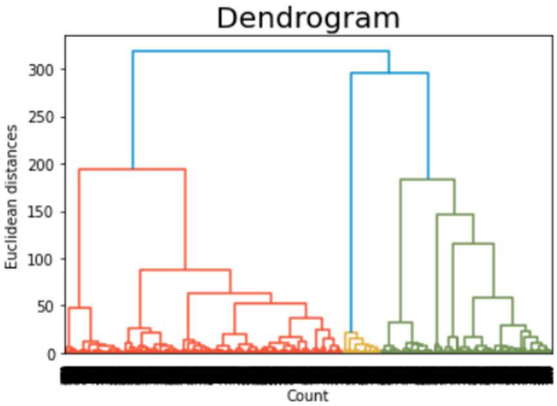

Fig 5. (A) Dendrogram for PCA

ResNet50 Representation

For applying Kmeans clustering algorithm on PCA and UMAP features of ResNet50 representation, elbow test and silhouette analysis is performed on both set of features. Hyperparameter tuning for UMAP features is shown in Fig 6, the optimal number of clusters for UMAP features is 4.

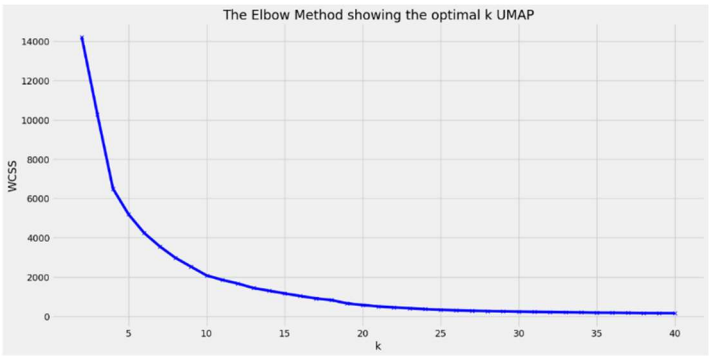

Fig 6 (A) Elbow Test for ResNet 50 UMAP features

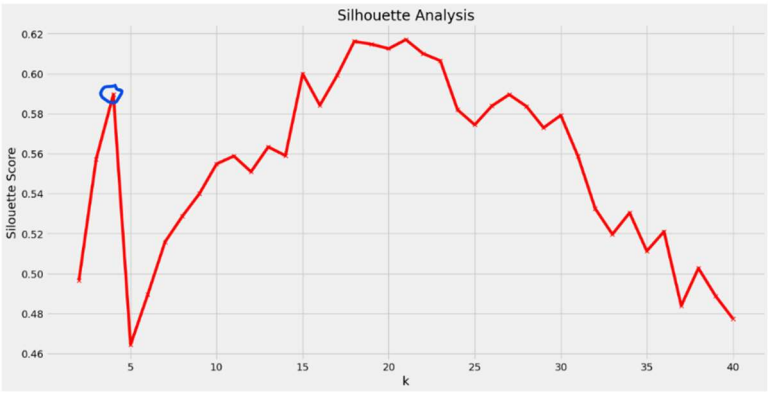

Fig 6 (B) Silhouette Analysis for ResNet50 UMAP features

For PCA features, the elbow test does not yield promising results as the cut-off point is not evident in Fig 7(A) but the silhouette score maximises at k=4 (Fig 7B). Therefore, the optimal number of clusters for PCA features is 4.

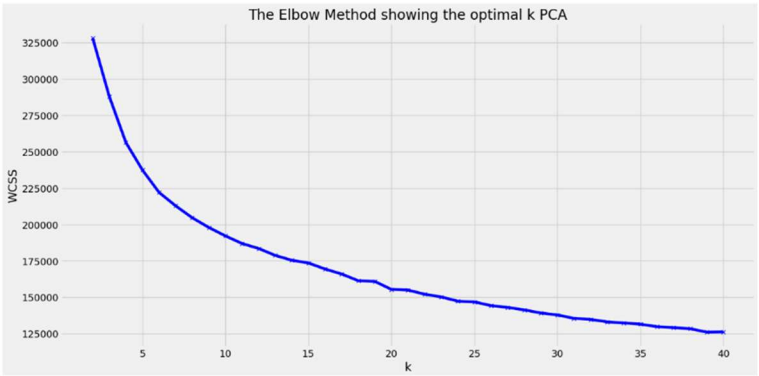

Fig 7 (A) Elbow Test for ResNet 50 PCA features

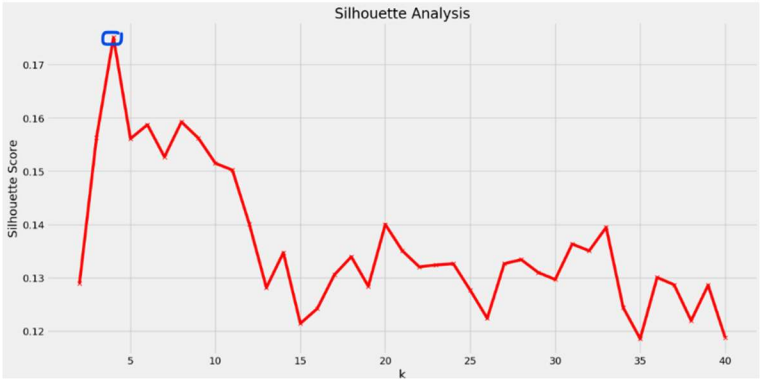

Fig 7 (B) Silhouette Analysis for ResNet50 PCA features

Similarly, for hierarchical clustering, the dendrograms for PCA and UMAP (Fig 8A and 8B) features return optimal number of clusters as 5 and 4 respectively.

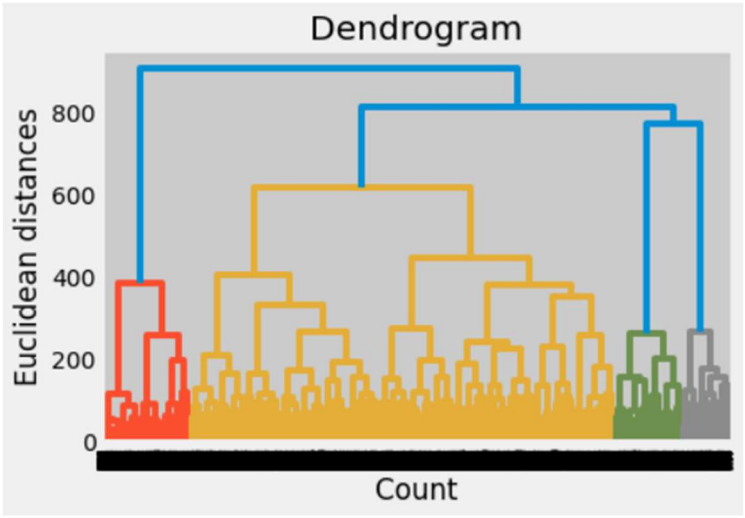

Fig 8 (A) Dendrogram for ResNet50 PCA features

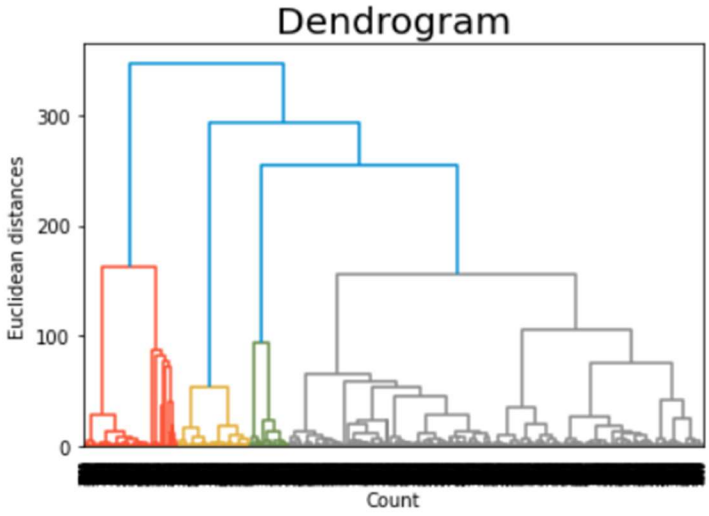

Fig8 (B) Dendrogram for ResNet50 UMAP features

Once the hyperparameters are computed, Kmeans and Hierarchical clustering models can be trained to group data into clusters. Then these models can be evaluated using different evaluation metrics.

Results

For PathalogyGAN the clustered data can be visualised using scatter plot. The scatter helps to visualize the data points as groups(clusters). The individual cluster configuration can be viewed using stacked column chart. It shows different tissue types that are part of a cluster and gives a high-level view of different classes of data that forms a cluster. Fig 9A and Fig 9B shows the clustering for UMAP features.

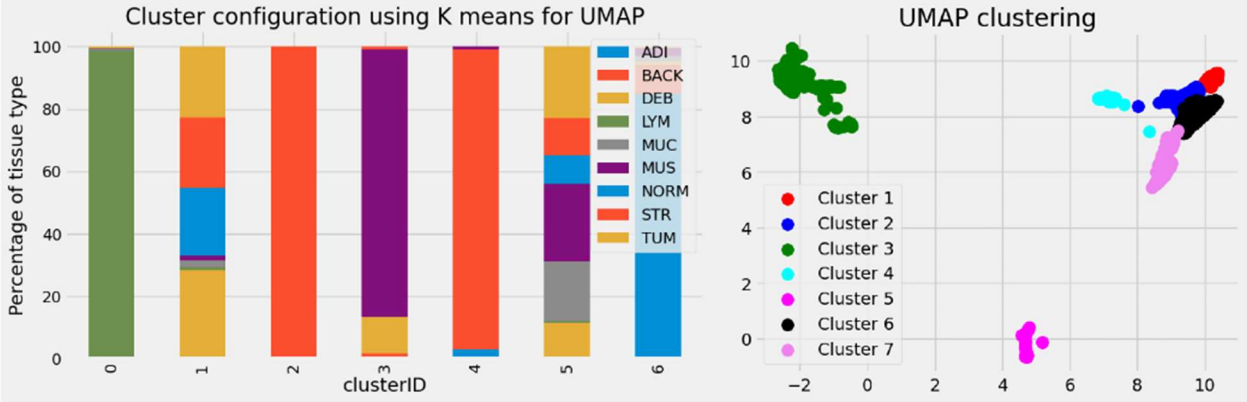

Fig 9A Stacked Column Chart for PathalogyGAN UMAP features

Fig 9B Scatter Plot for PathalogyGAN UMAP features

Fig 10A and Fig 10B helps us visualise the clusters and its configuration for PCA features.

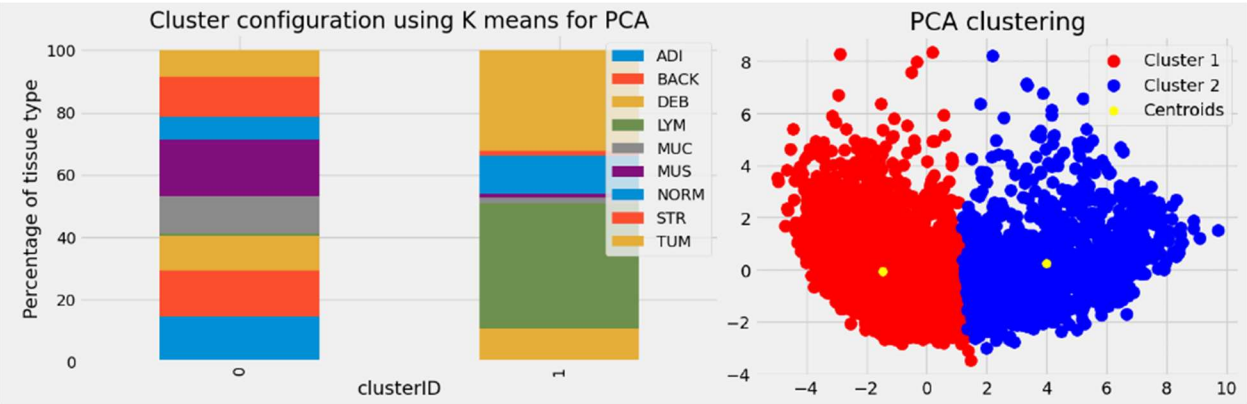

Fig 10A Stacked Column Chart for PathalogyGAN PCA features

Fig 10B Scatter Plot for PathalogyGAN PCA features

We can also look at the same type of visualization for Hierarchical clustering as well. Fig 11 and Fig 12 shows clustering configurations for both UMAP and PCA features.

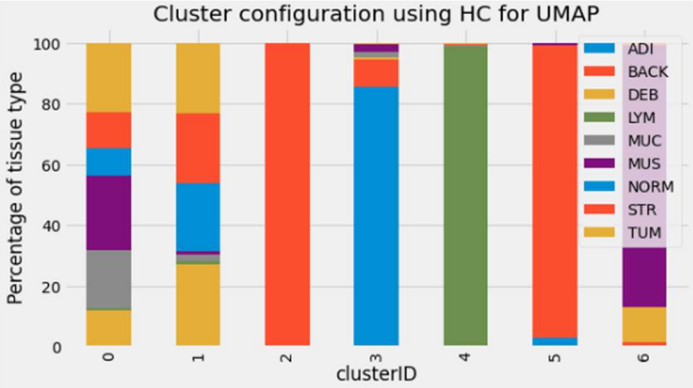

Fig 11A Stacked Column Chart for PathalogyGAN UMA features

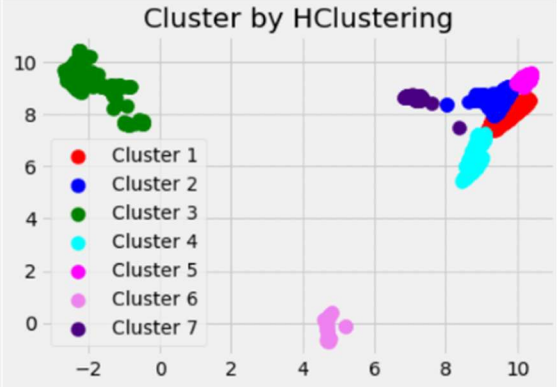

Fig 11B Scatter Plot for PathalogyGAN UMAP features

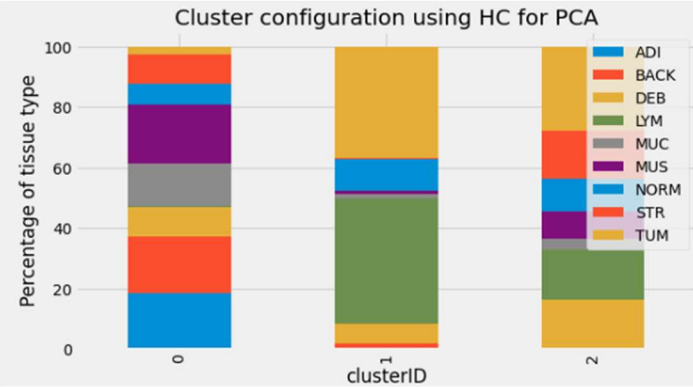

Fig 12A Stacked Column Chart for PathalogyGAN UMA features

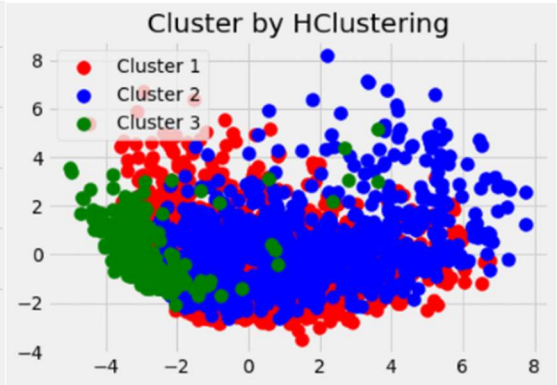

Fig 12B Scatter Plot for PathalogyGAN UMAP features

The same visualisations can also be plotted for ResNet50 representation.

Once we have visualised the clusters, now we need to evaluate the performance of the clustering model. For this purpose, we are using 4 different measures as described earlier. Table 1 shows the performance of Kmeans and Hierarchical clustering algorithms with both PCA and UMAP features for PathalogyGAN.

Kmeans And Hierchical Clustering of PathologyGAN on PCA and UMAP				
	HC		**Kmeans**	
Metrics	UMAP	PCA	UMAP	PCA
Silhouette	0.615047	0.137688	0.615687	0.307905
V-Measure	0.525014	0.220583	0.524363	0.203311
Davies-Bouldin	2.141611	1.937675	0.441953	1.381979
Calinski-Harabasz	7480.53175	475.002699	7551.96296	2050.912728

Table 1

As we can clearly observe in the Table 1 that for UMAP features, the overall result of Kmeans is better than Hierarchical clustering. Silhouette, V-Measure and Calinski-Harbasz score are approximately the

same, but the David-Bouldin score for Kmeans is much better. Also, both the algorithms fail to deliver acceptable performance when used to cluster PCA features.

Table 2 shows the performance of the Kmeans and HC for ResNet50 representation.

Kmeans And Hierchical Clustering of ResNet50 on PCA and UMAP				
	HC		Kmeans	
Metrics	UMAP	PCA	UMAP	PCA
Silhouette	0.579189	0.136208	0.583522	0.167112
V-Measure	0.548163	0.571993	0.526557	0.4793
Davies-Bouldin	0.707518	2.233807	0.671605	1.827544
Calinski-Harabasz	3349.654566	610.373415	3448.16682	718.817175

Table 2

For UMAP features, data points within clusters formed by Kmeans model are more similar. Also, the average similarity between cluster is lower for Kmeans model which is proved by the lower Davies-Bouldin score. As was the case with PathalogyGAN, both algorithms do not yield acceptable results when it comes to PCA features.

Conclusions

Looking at the results in the section above we can conclude the following: -

- For dimensionality reduction UMAP is superior to PCA as different models were able to cluster the UMAP features more efficiently than the PCA features.
- Hyperparameter tuning alone does not yield efficient clustering.
- Kmeans model have given better performance when compared to Hierarchical Clustering for PathalogyGAN representation.
- The 2-D scatter plot graphs show over overlapping which can be attributed to the multi-dimensional nature of the data.
- The Davies Bouldin score for both Kmeans and Hierarchical Clustering model on PCA features (for PathalogyGAN and ResNet50) goes out of acceptable range which clearly suggests that these models cannot be used.

References

[1] https://scikit-learn.org/

[2] V measure: A homogeneous and complete clustering –

https://towardsdatascience.com/v-measure-an-homogeneous-and-complete-clustering-ab5b1823d0ad

[3] Jupyter Notebook shared by Professor Ke Yuan