# COMPSCI5100 Machine Learning & Artificial Intelligence for Data Scientists (M)

# Case study 1: Model selection for clustering

**Group: 55**

**Group Member:**

**Name: Chengkun Yang**

**Name: Jinke Chen**

**Name: Yulin Xiao**

**Name: Lina Chen**

**Name: Yuge Wang**

## 1. Introduction

In this case study, the model selection for clustering was investigated. The significance of the model selection of clustering results in this case. With large and complex data sets, the time cost in model training can be long, and the accuracy can be different with different clustering methods. Thus, the selection of the best model among the algorithms that is fitted to the data can lead to a balance in the running speed and the accuracy of the outcome. With the most suitable model, the structure of the data can be better described. In the dataset processed, among 5000 colorectal cancer tissue patches, 9 tissue types were observed: Adipose (ADI), background (BACK), debris (DEB), lymphocytes (LYM), mucus (MUC), smooth muscle (MUS), normal colon mucosa (NORM), cancer-associated stroma (STR), and colorectal adenocarcinoma epithelium (TUM). Two representations were tested in this case: PathologyGAN and VGG16, with the dimensionality reduction methods of PCA and UMAP. The algorithms of Kmeans and Louvain were tested, and their performances were compared through the Silhouette Score and the V-measure score at the end.

## 2. Methods

In sum, K-means and Louvain were chosen as model training for this experiment, and Silhouette score and V measure score were chosen to assess the quality of the clustering solutions. Choosing K-means has these advantages: fast, simple, guaranteed convergence, and scales well with dataset size. Choosing Louvain clustering has these advantages: handles outliers, more informative, and scales well with dataset size and dimension.

1) K Means

K-Means algorithm is an unsupervised clustering algorithm. k-Means is essentially a data partitioning algorithm based on Euclidean distances, where dimensions with large means and variances will have a decisive influence on the clustering of the data.

First of all, a set of data should be divided into k categories, and k points should be randomly selected as the initial cluster centres. When the sample points are close to whichever initial clustering centre, then the sample points belong to whichever class. The horizontal and vertical coordinates of all sample points for each category are then summed and averaged to calculate a new coordinate point as the new cluster centre. This is repeated until all the sample points do not change categories.

As k increases, the sample division becomes finer and the degree of aggregation of each cluster gradually increases, then the error squared and SSE gradually becomes smaller. And when k reaches the true number of clusters, the degree of aggregation obtained by increasing k further will rapidly become smaller, so the drop in SSE will plummet.

2) Louvain algorithm

Louvain algorithm is a modularity-based graph algorithm model that, unlike normal modularity-based and modularity gain, runs very fast and performs clustering particularly well. The Louvain algorithm aims to optimize the modularity of the network by dividing it into densely connected groups of nodes. The Louvain algorithm consists of two steps. It first optimises modularity locally by finding small communities. It then aggregates the nodes in each small community and uses these aggregated nodes to construct a new network.

The Louvain algorithm flows by initially treating each vertex as a community, with the number of communities being the same as the number of vertices. Each vertex is merged with its neighbouring vertices in turn and their modularity gain is calculated to be greater than 0. If it is greater than 0, the node is placed in the community of the neighbouring node. Iterate through the second step until the algorithm is stable, i.e. the communities to which all vertices belong no longer change. Compress all nodes in each community into a single node, with the weights of the points within the community transformed into the weights of the new node's ring and the inter-community weights transformed into the weights of the new node's edges. Repeat the steps above until the algorithm is stable.

It is worth mentioning that the resolution in Louvain's algorithm affects the final number of classifications, so the step of selecting the resolution was added in this experiment.
This experiment started with an exploratory analysis, where the First 3 pricipal components of PathologyGAN's PCA feature were represented in a 3D graph and then compared. Secondly, the correlation model was poured into the experiment, and two-dimensional graphs (K-numbers and Scores) were drawn by analyzing the relationship between the number and the contour coefficients. Then finding and choosing k random points. Analysis of the resolution values in Louvain, plotting the number of K against the contour coefficients.

The plot chart shows the relationship between Louvain resolution and scores under both Silhouette and V measure evaluations. Furthermore, after checking out the number of clusters/cluster assignment counts, the number of clusters from the number of K-Means is 4 and from the number of Louvain is 2. In addition, getting the relationship forms about the Cluster index and the Number of members in K-means assignment counts and Louvain assignment counts. Then assessing goodness of fit by silhouette score and cluster homogeneities by V-measure. Then drawing the table about visualise tissue type percentage in two different clustering configurations.

## 3. Results

In this part, the clustering performance of K-means and Louvain Clustering using data from VGG16 is analyzed and compared using the Silhouette score and V-measure. The two dimensionality reduction methods: PCA and UMAP is also compared.

For training and testing. We used the PCA dimensionality reduction method to select 200 out of 5000 original data as training data.

The validation we choose cross validation, which is to repeatedly use data, divide the obtained sample data, and combine them into different training sets and test sets, use the training set to train the model, and use the test set to evaluate the quality of the model prediction. we considered to use the data that out of sample data, we will apply the model to data set in which model does not perform perfectly.

For the best classification in each situation, The K number of the K-means algorithm and the resolution of the Louvain algorithm should be determined. Comparing the Silhouette score and V-measure of each parameter, the following pictures can be obtained:

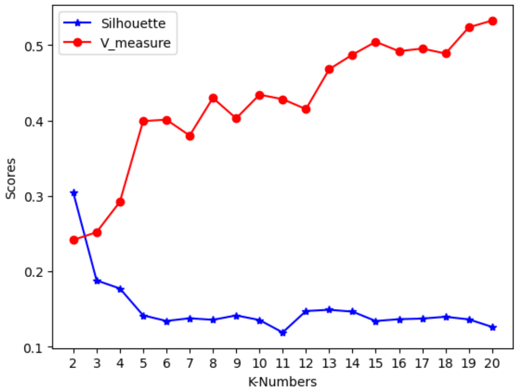

Figure 1 K number of pge pca

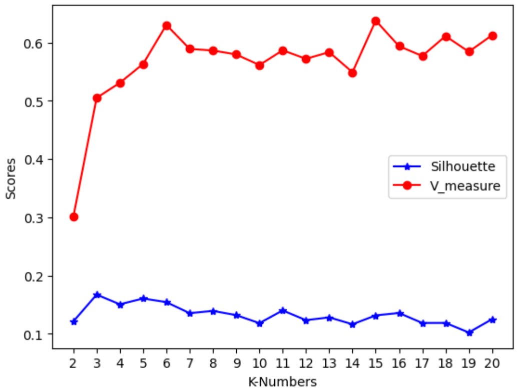

Figure 2 K number of vgg16 pca

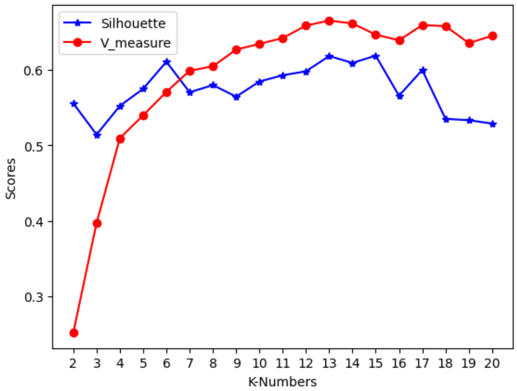

Figure 3 k number of pge umap

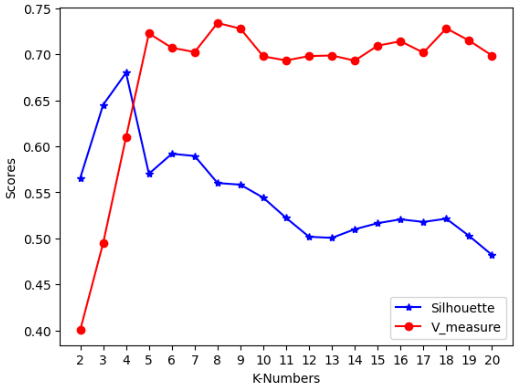

Figure 4 k number of vgg16 umap

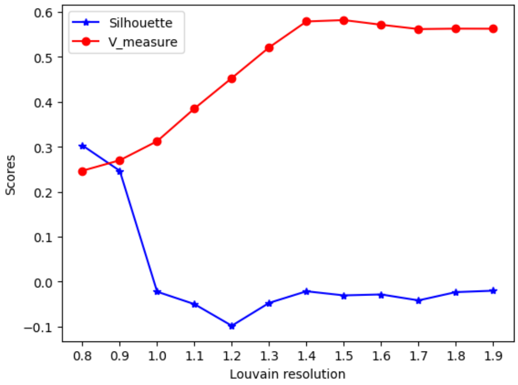

Figure 5 r number of pge pca

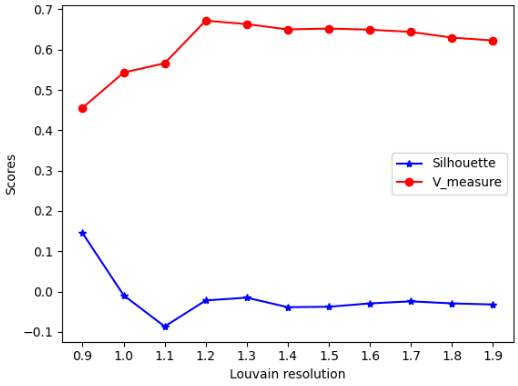

Figure 6 r number of vgg16 pca

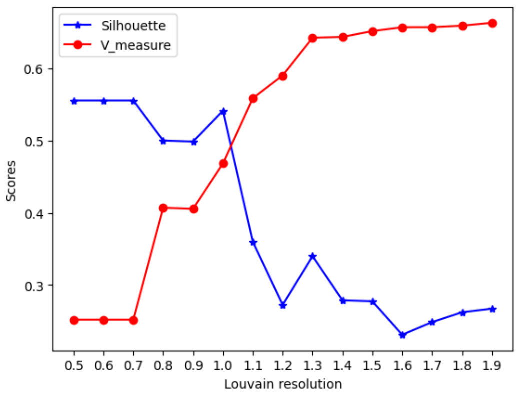

Figure 7 r number of pge umap

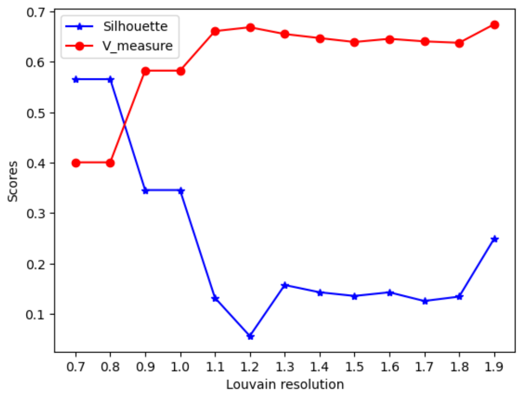

Figure 8 r number of vgg16 umap

Because that higher Silhouette Score means a dense and well separated results, and V-measure shows the balance of the homogeneity and completeness, the higher these two values are, the better performances is denoted. From the above pictures, the K-number and the resolution with the best performance can be determined.

Table 1 K number and resolution of datasets

| | Dimensionality reduction method | K-number | Resolution |
|---|---|---|---|
| PathologyGAN | PCA | 2 | 0.8 |
| | UMAP | 15 | 1.0 |
| VGG16 | PCA | 6 | 0.9 |
| | UMAP | 4 | 0.8 |

Using the parameters from the table above, the scores of the performance for each type of data and processing methods can be determined:

Table 2 scores of performances of datasets

| | Dimensionality reduction method | Clustering algorithm | Silhouette score | V-measure |
|---|---|---|---|---|
| PathologyGAN | PCA | K-means | 0.304120 | 0.241646 |
| | | Louvain | 0.303122 | 0.246820 |
| | UMAP | K-means | 0.618743 | 0.646719 |
| | | Louvain | 0.541286 | 0.468231 |
| VGG16 | PCA | K-means | 0.147305 | 0.602571 |
| | | Louvain | 0.14487 | 0.45582 |
| | UMAP | K-means | 0.679940 | 0.609898 |
| | | Louvain | 0.565764 | 0.400649 |

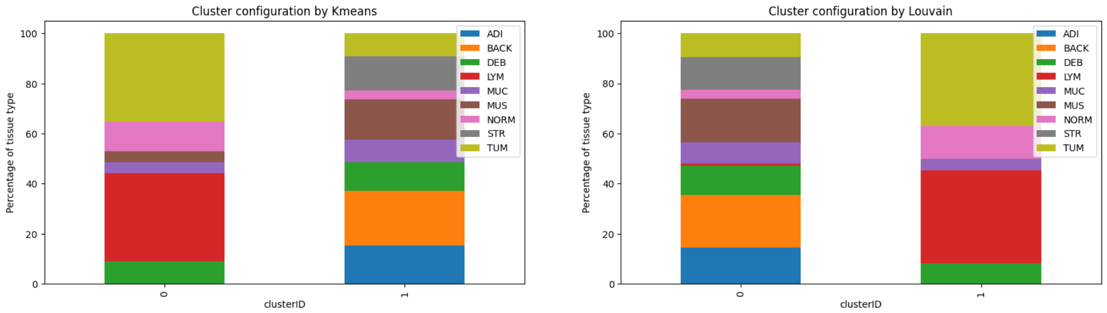

Figure 9 cluster configuration of pge pca

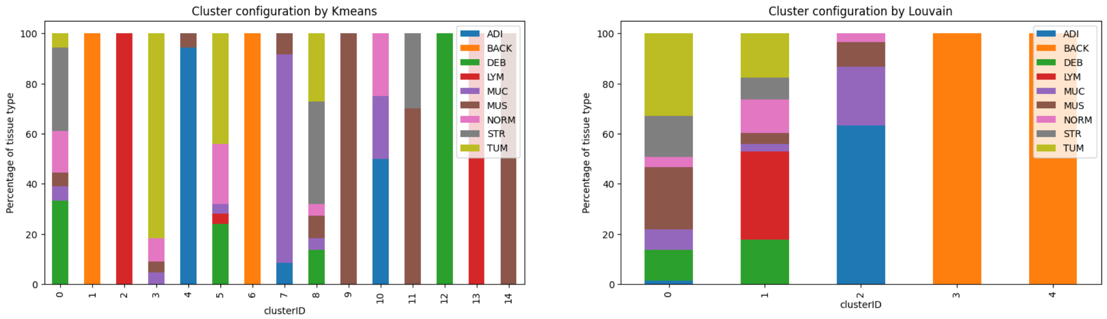

Figure 10 cluster configuration of pge umap

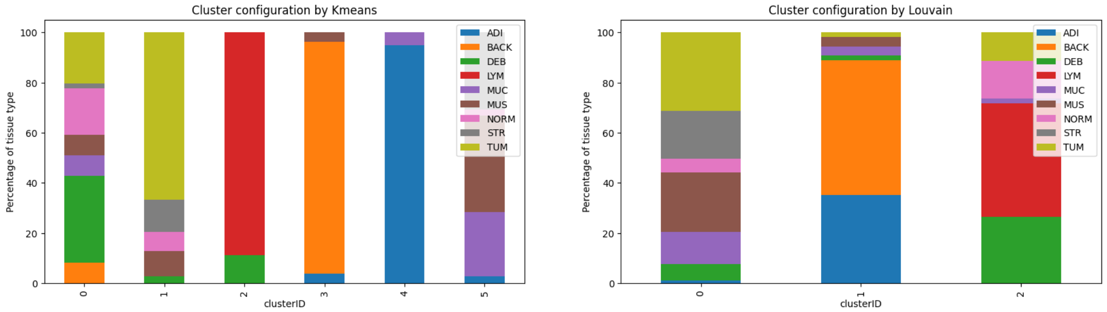

Figure 11 cluster configuration of vgg16 pca

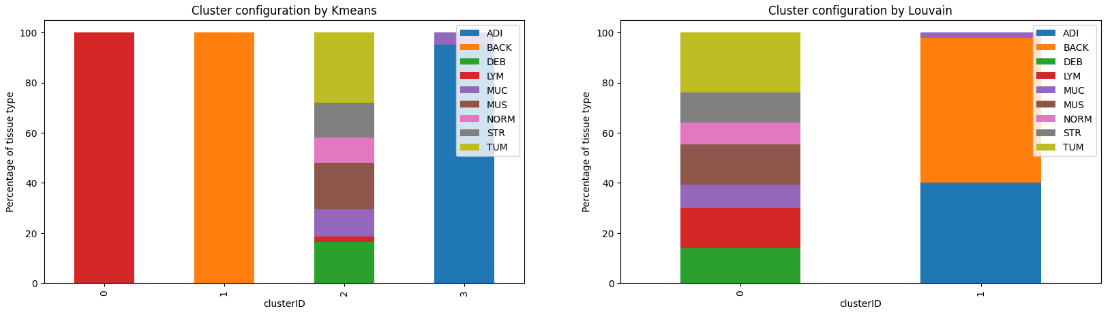

Figure 12 cluster configuration of vgg16 umap

From the table and pictures above, in general, it can be seen that for both PathologyGAN and VGG16, the UMAP method performs better than the PCA method. As for the algorithm, the K-means shows better results than the Louvain algorithm in all situations of this case. The K-means algorithm with the UMAP method shows the best results for both PathologyGAN and VGG16, and the Louvain algorithm with PCA methods lead to relatively the most unclear classifications.

## 4. Discussion

From the results, K-means and UMAP are significantly better than the Louvain Clustering and PCA for the processing of low-dimensional, low-noise and less variable type data like PathologyGAN and VGG16. From the main process of K-means we can see that its algorithmic idea is relatively simple and it converges quickly, so the clustering effect is better in this experiment. Louvain is an unsupervised algorithm (no input of community number or community size is required before execution), which is divided into two stages: modularity optimization and community aggregation, in which modularity is used to measure the quality of a community division, the larger the modularity the better the performance of the community division algorithm, so it does not work well in this experiment.

UMAP is a relatively flexible nonlinear dimensionality reduction algorithm that has emerged in recent years, and its theoretical basis is based on manifold theory and topological analysis. UMAP is more accurate and faster than other dimensionality reduction methods, for example, the computation time of t-SNE increases exponentially when the dimensionality of the data needs to be reduced, while UMAP increases linearly.

The purpose of PCA algorithm is to convert high-dimensional data to low-dimensional with less "information" loss, and to reduce the computational effort by analyzing the largest individual differences revealed by the principal components, which can also be used to reduce the number of variables in regression analysis and cluster analysis. Therefore, the essence of PCA is to find some projection directions so that the variance of the data in these projection directions is the largest, thus ignoring the variance in other directions, that is, PCA algorithm is better at high-dimensional data sets. However, the idea of UMAP is different from that of PCA. Its main purpose is to preserve the topological structure of its adjacent samples, that is, a finer local structure, rather than an overall structure. Therefore, for low-dimensional data samples, the UMAP algorithm will be better at processing them compared to the PCA algorithm.